ORSYP

Parce que votre Production Informatique est vitale

Dollar Universe v5.1

Manuel de l'interface commande

Dollar Universe Solutions pour Windows, Unix et VMS

Version 2.7

16 septembre 2005

COPYRIGHT

© Copyright ORSYP

Les composants suivants de DOLLAR UNIVERSE sont protégés par Copyright :

- le manuel d'installation,
- laissez-vous guider,
- le manuel d'administration,
- le manuel de référence,
- le manuel de l'interface Motif,
- le manuel de l'interface commande,
- le manuel de Global Control,
- le manuel de Business Views,
- le manuel de Web control,
- le manuel de DQM,
- tous les textes et titres des entrées et écrans du logiciel.

DOLLAR UNIVERSE, UNIVER$E GP et UNIVER$E DQM sont des marques déposées de ORSYP.

Les termes suivants sont la propriété d'ORSYP :

- Unité de Gestion ou UG.
- Uproc.

IBM®, AS/400™, OS/400™ sont des marques déposées de International Business Machines Corporation.

DEC®, VMS™ et OpenVMS™ sont des marques déposées de Hewlett Packard Corporation.

Windows NT™ et Windows 2000™ sont des marques déposées de Microsoft Corporation.

PATROL® est une marque déposée de BMC software.

SAP® Solutions est une marque déposée de SAP.

Java™ est une marque déposée de Sun Microsystems.

Tomcat™ est une marque déposée de Apache Software Foundation.

Oracle, Oracle8, PL/SQL, SQL*Plus sont des marques déposées de Oracle Corporation.

Les autres noms peuvent être des marques de leur propriétaire respectifs.

Table des matières

Spécifications générales des commandes 1

Présentation 1
- Les commandes de paramétrage 1
- Les commandes d'exploitation 2
- Les commandes de gestion de batch 2
- Les commandes de gestion de date 2
- Les commandes du CL 3
- Les variables 3

Spécifications générales 3
- Format d'une commande 3
- Utilisation des commandes 5

Les commandes de paramétrage 9

Génération des objets par un fichier d'ordres 9
- Description de la commande 9

Manipulation de l'administration 10
- Extraction et insertion 10
- Les sociétés 13
- Les noeuds 15
- Les types d'unités de gestion 18
- Les unités de gestion 19
- Les dépendances des unités de gestion 22
- Les domaines 23
- Les applications 25
- Les répertoires application 27
- Les répertoires unités de gestion 29
- Les profils utilisateurs 32
- Les utilisateurs 34

Définition des uprocs et sessions 36
- Les classes 36
- Les ressources 38
- Les uprocs 42
- Les sessions 56

Planification 61
- Les calendriers 61
- Les règles de planification 66
- Les tâches 69

Les commandes d'exploitation 83

Les objets de paramétrage 83
- Les ressources 83
- Les tâches 88

Le suivi d'exploitation 90

Les lancements prévus....90
Les événements....96
Les exécutions....100
Les outils de contrôle....105
Les automates....105
L'historique des exécutions....106
Le planning prévisionnel....109
Les statistiques....110

Gestion de batch 115

Présentation....115
Déclenchement de batch....115
Format....116
Description des items....116
Exemple....117
Surveillance de la durée d'exécution....118
uxsurjob....118
uxspvjob....119
uxalrjob....119

Gestion de date 121

Formatage et décalage de dates....121
Format....121
Description des items....122
Exemples....123
Formatage de date et décalage d'heure....124
Formats....124
Items....124
Exemples....125

Le CL d'exécution 127

Présentation du CL....127
Normes d'écritures du CL....127
Le code retour du CL....127
Les commandes du CL....128
Les commandes du CL....128
Ecriture de jalons....128
Affichage d'informations....129
Génération de messages....129
Passage de paramètres....130
Critères de gestion....130
Reconduction de variables....131

Annexes 135

Les items des commandes....135
Items d'environnement....135
Items d'entrée/sortie....135
Items d'identification d'objets....135
Autres items....136
Exemple d'utilisation des commandes du CL....140
Les jalons de reprise et les messages dans un script UNIX ksh....140
Les steps de reprise et les messages dans un script Unix CSH.....141

Release notes 143

Corrections 143
Corrections des commandes de paramétrage 143
Corrections des commandes d'exploitation 143
Corrections des commandes du CL d'exécution 143
Modification des commandes 143
Modification des commandes de paramétrage v5.0 SPRE0025 143
Modification des commandes de paramétrage v5.1 144
Modification des commandes d'exploitation 144
Modification des commandes de gestion de batch 144
Modification des commandes de dates 144
Nouvelles commandes 144
Nouvelles commandes de paramétrage 144
Nouvelles commandes d'exploitation 145
Nouvelles commandes de gestion de batch 145
Nouvelles commandes de dates 145
Nouvelles commandes du CL en v5.1 145
Commandes VMS 146

Index 149

Spécifications générales des commandes

Présentation

Les commandes de Dollar Universe permettent d'agir sur l'automate depuis un environnement extérieur (applicatif client, shell UNIX, scripts...). Elles se répartissent en cinq groupes présentés dans les cinq chapitres suivants de ce manuel :

- Les commandes de paramétrage qui permettent de définir ou de modifier le paramétrage de Dollar Universe,
- Les commandes d'exploitation qui permettent de consulter et d'intervenir sur l'exploitation en dehors des interfaces caractères ou graphiques,
- Les commandes de gestion des batchs afin de soumettre et de surveiller des exécutions,
- Les commandes de gestion de date,
- Les commandes du CL pour agir sur l'automate depuis le script d'une uproc.

Ces commandes sont unifiées pour les plate formes Windows, UNIX et VMS. L'utilisation de ces commandes est régie par des règles qui sont exposées dans la suite de ce chapitre.

Les commandes de paramétrage

Par thème, nous décrivons, ci-dessous, sous la forme d'un tableau, la liste des commandes de paramétrage disponibles (correspondant aux options de consultation et de mise à jour des fonctions interactives) :

Verbe : ux… Thème :	**add**	**upd**	**dup**	**dlt**	**shw**	**lst**	**tra**	**dst**	**ext**	**ins**	**rls**	**scv**
Administration									X	X		
Autorisations									X	X		
Société	X	X		X	X	X		X				
Noeud	X	X		X	X	X		X				
Type UG	X	X		X	X	X						
UG	X	X		X	X	X		X				
TIH	X	X		X	X	X		X				

Domaine	X	X		X	X	X		X				
Application	X	X		X	X	X		X				
Rep. Appli.	X	X		X	X	X		X				
Rep. UG.	X	X		X	X	X		X				
Profils	X	X		X		X						
Utilisateur	X	X		X	X	X		X				
Classe	X	X		X	X	X		X				
Ressource	X	X		X	X	X		X	X	X		
Règle	X	X		X	X	X			X	X		
Uproc	X	X	X	X	X	X	X	X	X	X	X	X
Session	X		X	X	X	X	X	X	X	X		
Calendrier	X	X	X	X	X	X	X	X	X	X		
Tâche	X	X	X	X	X	X	X	X	X	X		

Les commandes d'exploitation

Par thème, nous décrivons, ci-dessous, la liste des commandes d'exploitation disponibles :

Verbe **ux**… Thème :	**add**	**upd**	**dlt**	**shw**	**lst**	**pre**	**rsv**	**cnr**	**hld**	**rls**	**pur**	**rst**	**wak**	**str**	**end**
Ressource						X	X	X	X	X					
Tâche									X	X					
Lancement	X	X	X	X	X				X	X					
Evénement	X	X	X	X	X										
Contrôle					X						X	X			X
Automate				X									X	X	X
Hist. Exec.				X	X										
Planning p.					X										
Statistiques	X	X	X	X	X										

Elles permettent d'intervenir sur l'exploitation en cours.

Les commandes de gestion de batch

Deux commandes permettent de déclencher l'exécution d'un batch et d'en surveiller la durée d'exécution :

- **uxordre** (**uxord** sur VMS) déclenchement d'une tâche batch définie provoquée,
- **uxsurjob** (non disponible sous VMS) surveillance de la durée d'exécution d'une uproc et interruption de l'exécution au delà d'un temps limite.
- **uxspvjob** (non disponible sous VMS) surveillance de la durée d'exécution d'une uproc et exécution d'un script au delà d'un temps limite. Précis à la minute.
- **uxalrjob** (tous systèmes) surveillance de la durée d'exécution d'une uproc et exécution d'un script au delà d'un temps limite. Précis à la seconde.

Les commandes de gestion de date

Une commande permet de formater ou effectuer des décalages sur des dates :

- **uxdat** permet de modifier le format d'une date, de réaliser un décalage et d'obtenir le type ou le rang d'un jour dans le calendrier.
- **uxtim** permet de convertir une date (connue ou non) dans un format donné et/ou d'y ajouter un décalage horaire..

Les commandes du CL

Les commandes **uxset** permettent d'agir sur l'automate depuis les scripts des uprocs :

- **uxset msg** : envoi de messages dans la trace automate de l'exécution à des fins de diagnostic.
- **uxset step** : pose de jalons pour suivre l'exécution d'un script.
- **uxset parm** : passage de paramètres, au sein d'une session, d'une uproc à sa suivante.
- **uxset mgt** : définition des critères de gestion d'une uproc.
- **uxset var** : reconduction de la valeur des variables d'une uproc au sein d'une session.
- **uxset info** : affichage d'informations supplémentaires dans le suivi des exécutions.

Les variables

Un ensemble de variables contextuelles à l'exécution des uprocs et à la configuration de Dollar Universe est disponible dans le CL, il peut être utilisé pour tester, par exemple, un compte de soumission, le noeud ou l'unité de gestion d'exécution, etc. Voir manuel de référence, paragraphe "Paramètres d'exécution".

Les variables d'environnement utilisées doivent être connues du serveur d'IO et donc définies :

- Pour les systèmes Windows, dans le fichier %UXEXE%\<SOCIETE>.DEF,
- Pour les systèmes UNIX, dans les fichiers $UXMGR/uxsetenv, $UXMGR/uxsetenv_csh et $UXMGR/uxsetenv_ksh,
- Pour les systèmes VMS, dans les fichiers UXEXE:U_BATCH.COM ou UXEXE:U_ORD.COM.

Spécifications générales

Pour que la commande puisse être exécutée, il faut que les serveurs d'I/O de la société soient démarrés.

Format d'une commande

Une commande se découpe en cinq parties principales :

- Désignation du verbe de la commande utilisée, par exemple uxadd,
- Définition du thème de travail pour la fonction activée, par exemple UPR,
- Sélection de l'environnement de travail : société, noeud, espace et éventuellement unité de gestion,
- Sélection des enregistrements de travail, avec positionnement des informations souhaitées,
- Définition des fichiers contenant des informations en "entrée" de la fonction (input=fichier) ou de recueil des informations en "sortie" (output=fichier).

Verbe

Le verbe de la commande doit toujours être cité en premier. Si le répertoire de travail courant n'est pas le répertoire UXEXE, celui-ci doit être précisé dans la commande. Par exemple sous UNIX :

```
$UXEXE/uxlst UPR
```

Pour obtenir la liste des uprocs de l'espace par défaut.

Thème

Le thème doit toujours être cité en second à la suite du verbe de la commande, les items étant décrits par la suite dans un ordre quelconque.

```
$UXEXE/uxlst UPR
```

Dans l'exemple ci-dessus le thème est UPR. Il indique que la commande porte sur les uprocs.

Environnement

Les informations permettant cette sélection sont facultatives, les défauts considérés sont :

- SOC = S_SOCIETE (si le symbole est défini, sinon l'information est **obligatoire**),
- NODE = S_NOEUD (généré avec S_SOCIETE), les commandes du CL, Les commandes de paramétrage, d'exploitation, de déclenchement de batch ne pouvant être effectuées que localement, le code du noeud doit être celui du noeud local.
- EXP (espace d'exploitation), SIM pour simulation, INT pour intégration et APP pour application. L'espace indiqué en insertion doit être le même qu'en extraction. Ces commandes ne peuvent pas être utilisées pour transférer du paramétrage d'un espace dans l'autre. Si l'espace n'est pas indiqué c'est la valeur de la variable ou du symbole S_ESPEXE qui sera pris en compte.

Items

Pour chaque commande, une liste d'items est détaillée :

- Les items obligatoires sont présentés en gras,
- "*" (astérisque) souligne la possibilité de saisir une valeur générique pour l'item concerné.

Fichiers d'entrée / sortie

Les items input et output sont principalement utilisés avec les commandes d'extraction et d'insertion **uxext** et **uxins** proposées par Dollar Universe afin de permettre une interface facilitée avec un produit de télédistribution par exemple.

L'item output peut également être utilisé par les commandes **uxshw** ou **uxlst** pour recueillir les informations dans un fichier à la place de l'affichage à l'écran (valeur par défaut).

Un même fichier de sortie peut être utilisé pour un ensemble de commandes **uxext** (pouvant porter sur des objets différents), les données extraites étant ajoutées en fin de fichier à celles déjà extraites par les commandes précédentes.

Il est cependant recommandé de ne **pas** utiliser le même fichier de sortie pour des objets appartenant à des espaces différents, dans ce dernier cas l'utilisation de caractères génériques sur les version d'objets (uprocs, session et tâches), pourrait générer des confusions. Par exemple, si le même fichier contient le résultat des deux commandes suivantes :

```
uxext UPR APP UPR="*" VUPR="*" output=fichier
uxext UPR INT UPR="*" VUPR="*" output=fichier
```

Si le même couple (uproc, version) existe dans les deux espaces, et comme l'item REPL n'est pas indiqué seule l'uproc version de l'espace application sera inséré dans le fichier.

Si l'item REPL était indiqué dans le deuxième commande, seule l'uproc version de l'espace intégration persistera dans le fichier (en écrasant la définition donnée en espace application).

Code retour

Deux codes retour sont retournés par les commandes :

- 0 : la commande a été exécutée correctement,
- 1 : la commande n'a pas pu être exécutée correctement, un message complète ce code retour pour expliquer la raison de l'échec : erreur de syntaxe, serveur d'I/O absent ...

Utilisation des commandes

Les commandes de paramétrage, citées dans le tableau ci-dessus peuvent être utilisées d'une part sous la ligne de commande du système d'exploitation ou dans une procédure et d'autre part dans un fichier d'ordres utilisé par la commande de chargement **uxload**.

Les commandes d'exploitation citées dans le second tableau ainsi que les commandes de gestion de batch peuvent être utilisées sous la ligne de commande du système d'exploitation ou dans une procédure.

Les commandes du CL ne peuvent être utilisées que dans un CL associé à une uproc. Elles ne peuvent s'exécuter que dans le contexte fourni par Dollar Universe.

Les particularités liées aux différents modes d'utilisation sont décrites ci-après.

Règles générales

Pour plus de lisibilité des commandes le caractère "\" peut être utilisé pour indiquer le passage à la ligne suivante.

La virgule est systématiquement interdite ainsi que tout autre caractère non cité. Les "+" et "-" ne sont autorisés dans les codes des objets que pour règles.

Tous les mots clés peuvent être saisis en minuscules ou en majuscules indifféremment, seule l'utilisation différenciée de majuscules et de minuscules doit être respectée dans les verbes et les zones de saisie.

Les objets de Dollar Universe doivent tous être saisis en majuscules sauf pour les noeuds et les utilisateurs où la casse doit être respectée.

Quelle syntaxe pour quel système d'exploitation

Considérant Windows, UNIX ou VMS, chacun de ces systèmes d'exploitation utilise une liste de caractères spécifiques qui peuvent être utilisés dans une commande de Dollar Universe.

A cette liste, peut être ajoutée la liste des caractères qui sont un sens particulier pour les commandes de Dollar Universe. Ces caractères sont : espace = ( ) { }.

Ces caractères doivent être protégés afin d'être transmis aux commandes de Dollar Universe sans avoir été interprétés par le système d'exploitation. Le type de protection dépend du système d'exploitation.

Restrictions concernant la commande uxload

Les commandes de paramétrage peuvent être utilisées dans un fichier d'ordres, ce fichier pouvant par la suite être exécuté par la commande **uxload**. Dans ce cas de figure la restriction sur les guillemets ne s'applique pas.

Caractères spéciaux sous Windows

Espace + - ( ) sont des caractères spécifiques.

Un seul type de protection peut être utilisé : l'anti-slash + double guillemet, par exemple : \"texte ou champ texte\".

Exemples

Correct :

uxadd MU MU=CENTER LABEL=\"Siège société\" TNODE=%S_NOEUD%

uxupd MU MU=CENTER LABEL=\" Siège société (M)\"

uxupd UPR EXP UPR=D_LOAD_FIL TERMINS add TERMNO=02 TERM=D_LOAD_FIL STATUS=ABNORMAL PDATE=\"(-1,,)\"

uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM=\"=C01 AND =C02\"

Incorrect :

uxadd MU MU=CENTER LABEL= Siège société TNODE=%S_NOEUD%

uxadd MU MU=CENTER LABEL="' Siège société "' TNODE=%S_NOEUD%

Caractères spéciaux sous UNIX

Espace + - ( ) { } [ ] sont des caractères spécifiques.

Deux types de protection peuvent être utilisés :

- L'anti-slash + guillemet (si l'espace est le seul caractère spécifique), par exemple : \"texte ou champ texte\",
- L'apostrophe + guillemet, par exemple : "'texte (ou champ texte)"'.

Exemples

Correct :

uxadd MU MU=CENTER LABEL=\" Siège société \" TNODE=$S_NOEUD

uxupd MU MU=CENTER LABEL=\" Siège société \(M\)\"

uxadd MU MU=CENTER LABEL="' Siège société (M)"' TNODE=$S_NOEUD

uxupd UPR EXP UPR=D_LOAD_FIL TERMINS add TERMNO=01 TERM=D_LOAD_FIL STATUS=ABNORMAL PDATE="'(-1,,)"'

uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM="'=C01 AND =C02"'

uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM=\"=C01 AND =C02\"

Incorrect :

uxadd MU MU=HEADQUART LABEL=" Siège société " TNODE=$S_NOEUD

uxadd MU MU=HEADQUART LABEL= Siège société TNODE=$S_NOEUD

uxadd MU MU=CENTER LABEL=\" Siège société (M)\" TNODE=$S_NOEUD

uxupd UPR EXP UPR=D_LOAD_FIL TERMINS add TERMNO=01 TERM=D_LOAD_FIL STATUS=ABNORMAL PDATE=\"(-1,,)\"

uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM="=C01 AND =C02"

Attention, certains interpréteurs de commandes peuvent avoir d'autres particularités. Ces exemples ont été testés avec le ksh et le csh.

Caractères spéciaux sous VMS

+ - ( ) sont des caractères spécifiques.

Un seul type de protection peut être utilisé : le triple guillemet, par exemple : """texte (ou champ texte)""".

Exemples

Correct :

uxadd MU MU=CENTER LABEL=""" Siège société (M)""" TNODE=TEST

uxupd UPR EXP UPR=D_LOAD_FIL TERMINS add TERMNO=02 TERM=D_LOAD_FIL STATUS=ABNORMAL PDATE="""(-1,,)"""

uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM="""=C01 AND =C02"""

Incorrect :

uxadd MU MU=CENTER LABEL=" Siège société M" TNODE=TEST

uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM="=C01 AND =C02"

Respect de la casse

VMS possède deux particularités supplémentaires :

- lorsque la commande est transmise au système d'exploitation, la casse des arguments est automatiquement modifiée en majuscules,
- lorsqu'une commande "foreign" (cas des commandes de Dollar Universe) est transmise au système d'exploitation, la casse des arguments est parfois (cela dépend de la version du système d'exploitation) automatiquement modifiée en minuscules.

Les objets de Dollar Universe peuvent être écrits en majuscules (par défaut) ou minuscules sauf le nom de nœud Dollar Universe et le compte utilisateur qui doivent respecter la casse avec laquelle ils ont été définis (la définition d'un nom de nœud doit être exactement la même sur les différents nœuds).

Pour protéger la casse, le nom de nœud doit toujours être protégé dans toutes les commandes utilisant les items NODE et TNODE, le compte utilisateur doit toujours être protégé dans toutes les commandes utilisant l'item USER.

La protection est réalisée par des guillemets, par exemple : "Noeud".

Il est également recommandé de protéger tous les arguments par des guillemets.

Exemples

uxadd NODE TNODE="Node" LABEL="""Test machine""" DIR="T"

uxshw NODE TNODE="Node"

Il est recommandé d'utiliser la commande **uxshw node** pour vérifier la création correcte du nœud après avoir exécuté une commande **uxadd node**.

Format des dates

Toutes les dates saisies ou affichées répondent, depuis la version 4.3 de Dollar Universe à la règle ci-dessous.

La variable U_FMT_DATE permet de définir le format souhaité des dates. Sa valeur doit être construite à partir des composants suivants :

- YYYY : année sur 4 caractères,
- MM : numéro de mois de 01 à 12 (sur deux caractères),
- DD : jour dans le mois de 01 à 31 (sur deux caractères),
- Un séparateur qui peut être choisi parmi les valeurs caractères suivants : / (slash) ou : (deux points) ou - (tiret) ou pas de séparateur. Le séparateur, s'il est utilisé, ne peut servir qu'à séparer les trois composantes de la date et non à les encadrer.

Par exemple si on souhaite exprimer la date du 21 juin 1997 :

- Si U_FMT_DATE vaut DDMMYYYY, la date affichée ou saisie sera le 21061997,
- Si U_FMT_DATE vaut YYYY-DD-MM, la date affichée ou saisie sera le 1997-21-06.

Si la variable U_FMT_DATE n'est pas valorisée alors sa valeur par défaut est DD/MM/YYYY en Français, MM/DD/YYYY en Anglais.

Les commandes de paramétrage

Génération des objets par un fichier d'ordres

Description de la commande

Fonction

Cette fonction permet de générer du paramétrage Dollar Universe, indépendamment des interfaces caractère ou graphique, à partir d'un fichier d'ordres descriptif des objets à mettre en oeuvre.

Les ordres pris en compte par cette commande portent sur les verbes **uxadd** (ajouter), **uxupd** (modifier) et **uxdlt** (supprimer).

Tous les objets de définition du paramétrage de Dollar Universe peuvent être manipulés par cette commande : tables d'administration, uprocs, ressources, sessions, calendriers, règles et tâches.

Format

Aucune définition d'environnement n'est nécessaire à cette commande puisque la définition de l'environnement est portée directement par les commandes de chargement de paramétrage.

uxload

Item	Format	Défaut
VERIFY | EXECUTE DEF=file_name **INPUT=file_name** OUTPUT=file_name		

Description des items

VERIFY : Vérification de la syntaxe du fichier d'ordres. Utilisation incompatible avec EXECUTE.

Permet la vérification seule du fichier d'ordres sans charger aucun objet dans Dollar Universe. Le résultat de cette vérification est obtenu dans le fichier résultat passé en paramètre.

EXECUTE : Vérification et exécution du fichier d'ordres. Utilisation incompatible avec VERIFY.

Permet la vérification du fichier d'ordres ainsi que l'exécution des commandes de ce fichier. Le résultat de cette vérification est obtenu dans le fichier résultat passé en paramètre.

L'un des deux items VERIFY ou EXECUTE est obligatoire.

DEF : Fichier des valeurs par défaut.

Ce fichier permet une saisie unique de valeurs qui seraient répétées dans des commandes similaires.

Son utilisation est facultative, s'il n'est pas utilisé, les valeurs par défaut des différentes commandes décrites ci-après seront celles mentionnées dans cette documentation (issues du fichier cmd_u.def).

Sa syntaxe est la suivante : PARAMETRE=VALEUR où :

- PARAMETRE est le nom qui sera utilisé dans le fichier d'ordre,
- VALEUR est la valeur par lequel PARAMETRE sera remplacé lors de l'exécution de la commande **uxload**.

INPUT : Fichier d'ordres. Contient toutes les commandes qui doivent être exécutées. Syntaxe particulière :

- une ligne de commentaire débute par un "/" en première position,
- la création d'un objet complet nécessitant parfois la répétition de son identifiant (uproc ou tâche par exemple), on peut utiliser le raccourci "()". L'utilisation de ce raccourci signifie que l'ordre courant utilise les identifiants de l'ordre immédiatement précédent, éventuellement complétés des identifiants suivant les caractères (). Par exemple les trois lignes suivantes :

uxadd UPR APP UPR=IU_TEST VUPR=001 ....

uxudp UPR APP UPR=IU_TEST VUPR=001 DEPCON add....

uxudp UPR APP UPR=IU_TEST VUPR=001 RESCON add....

peuvent être remplacées par :

uxadd UPR APP UPR=IU_TEST VUPR=001 ....

uxudp () DEPCON add...

uxudp () RESCON add...

- L'utilisation d'une valeur définie dans le fichier des défauts (DEF) se fait en utilisant la syntaxe suivante : ~PARAMETRE~ (le nom du paramètre entouré de deux tildes). La commande **uxload** substituera la valeur au nom lors de l'exécution.

Le fichier est obligatoire.

OUTPUT : Fichier résultat d'exécution de la commande. Le fichier est obligatoire.

Son utilisation dirige dans ce fichier toutes les informations et tous les messages qui auraient été affichés à l'écran.

Remarque : Cette commande génère un fichier log nommé uxload.log et situé dans le répertoire du log général Dollar Universe. Il contient tous les messages d'information ou d'erreur liés à la vérification de la syntaxe et de la cohérence du fichier d'ordres ainsi que les messages issus de l'exécution de cette commande.

Manipulation de l'administration

Extraction et insertion

Les commandes

uxext ADM : extrait une table d'administration ou une partie de cette table.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD

TSOC=	6c *	
TNODE=	10c *	
DOM=	1c *	
APPL=	2c *	
CLASSE=	6c *	
TYPUG=	1c *	
UG=	10c *	
TIH		
USER=	12c *	
REPERT		
REPL		
OUTPUT=file_name		
NOVERIF		

Exemple :

```
uxext ADM APPL=B* OUTPUT="c:\temp\adm.ext"
```

uxins ADM : insère des informations issues d'une extraction dans les tables d'administration correspondantes pour la même version de Dollar Universe.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TSOC=	6c *	
TNODE=	10c *	
DOM=	1c *	
APPL=	2c *	
CLASSE=	6c *	
TYPUG=	1c *	
UG=	10c *	
TIH		
USER=	12c *	
REPERT		
REPL		
INPUT=file_name		

Exemple :

```
uxins ADM APPL=* INPUT="c:\temp\adm.ext"
```

Description des items

TSOC : table des sociétés. Saisie d'un code société (valeur générique acceptée).

Format : 6 caractères alphanumériques majuscules.

TNODE : table des noeuds. Saisie d'un code noeud (valeur générique acceptée).

Format : 10 caractères alphanumériques.

DOM : table des domaines. Saisie d'un code domaine (valeur générique acceptée).

Format : 1 caractère alphanumérique.

APPL : table des applications et des répertoires d'application.

Saisie d'un code application (valeur générique acceptée),

table des applications si REPERT n'est pas mentionné,

table des répertoires d'applications si REPERT est indiqué; dans ce cas, les répertoires de toutes les applications renseignés dans la table sont extraits (les répertoires de toutes les applications contenus dans le fichier d'entrée sont insérés).

Format : 2 caractères alphanumériques.

CLASSE : table des classes d'incompatibilités. Format : 6 caractères alphanumériques.

Saisie d'une classe d'incompatibilités (valeur générique acceptée).

TYPUG : table des types d'unité de gestion. Format : 1 caractère alphanumérique.

Saisie d'un code type d'unité de gestion (valeur générique acceptée).

UG : table des unités de gestion et des répertoires d'unités de gestion.

Saisie d'un code unité de gestion (valeur générique acceptée),

table des unités de gestion si REPERT n'est pas mentionné,

table des répertoires d'unités de gestion si REPERT est indiqué. Toutes les applications sont considérées dans l'extraction ou l'insertion.

Format : 10 caractères alphanumériques.

TIH : table du réseau des unités de gestion.

L'extraction se fait de manière globale (pas de saisie de valeur et donc pas de sélection possible).

USER : tables des utilisateurs. Saisie d'un nom d'utilisateur (valeur générique acceptée).

Format : 12 caractères alphanumériques.

Remarque Les items précédents sont incompatibles (extraction d'une seule table à la fois) hormis APPL et UG utilisés avec REPERT pour indiquer la table des répertoires d'unités de gestion. L'un au moins de ces items est requis.

REPERT : répertoires d'application ou d'unité de gestion.

Se combine avec APPL et/ou UG pour déterminer la table à traiter.

REPL : Option de remplacement.

Indique si l'enregistrement extrait (inséré) annule et remplace l'enregistrement déjà existant dans le fichier de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signalé par un message pour chaque occurrence).

OUTPUT : fichier de sortie pour la commande **uxext.** Le nom du fichier est **obligatoire**.

INPUT : fichier d'entrée pour la commande **uxins.** Le nom du fichier est **obligatoire**.

NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

Exemples

```
uxext ADM APPL=BK output=c:\temp\DOC.ext
```

extrait l'application BK de la société courante et archive les données dans le fichier **c:\temp\DOC.ext**.

```
uxins ADM APPL=* input=c:\temp\DOC.ext
```

insère toutes les applications du fichier **c:\temp\DOC.ext** dans l'environnement courant (société et noeud).

```
uxext ADM UG=S* output=c:\temp\DOC.ext
```

```
uxext ADM UG=A* output=c:\temp\DOC.ext
```

extrait toutes les unités de gestion de type S et A et archive les données dans le fichier **c:\temp\DOC.ext**.

```
uxins ADM UG=* input=c:\temp\DOC.ext
```

insère toutes les unités de gestion du fichier **c:\temp\DOC.ext** dans l'environnement courant (société et noeud).

Les sociétés

Les commandes

uxadd SOC : crée une société dans la table des sociétés.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TSOC=	**6c**	
LABEL=	**20c**	
MASTER=	10c	
LOCK | UNLOCK		UNLOCK
TPU | EDT		
DIR=	**60c**	

Exemple :

```
uxadd SOC TSOC=EXAMPL LABEL=\"new company\" DIR="c:\EXAMPL"
```

uxupd SOC : modifie les caractéristiques d'une société.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TSOC=	**6c ***	
LABEL=	20c	
MASTER=	10c	
LOCK | UNLOCK		
TPU | EDT		
DIR=	60c	

Exemple :

```
uxupd SOC TSOC=EXAMPL DIR="d:\EXAMPL"
```

uxshw SOC : affiche les caractéristiques d'une société.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TSOC=	**6c ***	
OUTPUT=file_name		

Exemple :

```
uxshw SOC TSOC=*
```

uxdlt SOC : supprime une société de la table sociétés.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TSOC=	**6c ***	

Exemple :

```
uxdlt SOC TSOC=EXAMPL
```

uxlst SOC : liste la table des sociétés.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TSOC=	6c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst SOC
```

uxdst SOC : distribue la table des sociétés.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst SOC TMU=S_URANUS
```

Description des items

TSOC : Code société. Format : 6 caractères alphanumériques majuscules.

Saisie d'un code société (valeur générique acceptée sauf pour **uxadd**).

LABEL : Libellé descriptif de la société. Format : 20 caractères alphanumériques.

LOCK : Verrouillage des uprocs et des sessions lors des opérations de transfert ou de distribution

Défaut : UNLOCK.

MASTER : Noeud maître auquel est rattaché la société. Format : 10 caractères alphanumériques.

Défaut : noeud courant.

TPU : Editeur utilisé pour le CL des uprocs et les logs.

Sous VMS uniquement. Défaut : EDT.

DIR : Répertoire d'installation de la société.

Obligatoire pour la commande **uxadd**. Format : 60 caractères alphanumériques.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

OUTPUT : Stockage des résultats. Le nom de fichier est obligatoire pour la commande **uxext**.

Si l'item n'est pas renseigné la sortie se fait à l'écran,

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxupd SOC TSOC=EXAMPL LABEL=\"Société de test\"
```

modifie le label de la société EXAMPL.

```
uxshw SOC TSOC=*
```

affiche la définition des sociétés.

Les noeuds

Les commandes

uxadd NODE : crée un noeud dans la table des noeuds.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TNODE=	**10c**	
LABEL=	**20c**	
MONI | NOMONI		NOMONI
DEV | NODEV		DEV
PROD | NOPROD		PROD
DIR=	60c	
XDIR=	60c	
SDIR=	60c	
IDIR=	60c	
ADIR=	60c	
DDIR=	60c	
PDIR=	60c	

Exemple :

```
uxadd NODE TNODE=Asterix LABEL=\"Test machine\" DIR="C:\UNIV51"
XDIR="C:\UNIV51\exp\data" SDIR="C:\UNIV51\sim\data"
IDIR="C:\UNIV51\int\data" ADIR="C:\UNIV51\app\data"
DDIR="C:\Univer$e\5.0.0\exec" PDIR="C:\Univer$e\5.0.0\exec"
```

uxupd NODE : modifie les caractéristiques d'un noeud.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD

TNODE=	**10c ***	
LABEL=	20c	
MONI | NOMONI		
DEV | NODEV		
PROD | NOPROD		
DIR=	60c	
XDIR=	60c	
SDIR=	60c	
IDIR=	60c	
ADIR=	60c	
DDIR=	60c	
PDIR=	60c	

Exemple :

```
uxupd NODE TNODE=Asterix LABEL=\"Test node\"
```

uxshw NODE : affiche les caractéristiques d'un noeud.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TNODE=	**10c ***	
OUTPUT=file_name		

Exemple :

```
uxshw NODE TNODE=A*
```

uxdlt NODE : supprime un noeud de la table des noeuds.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TNODE=	**10c ***	

Exemple :

```
uxdlt NODE TNODE=Obelix
```

uxlst NODE : liste la table des noeuds.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TNODE=*	10c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst NODE OUTPUT="c:\temp\LstNode.txt"
```

uxdst NODE : distribue la table des noeuds.

Item	Format	Défaut
SOC=	6c	S_COMPANY

NODE=	10c	S_NOEUD
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst NODE TMU=S
```

Description des items

TNODE : Code noeud. Saisie d'un code noeud (valeur générique acceptée sauf pour **uxadd**).

Format : 10 caractères alphanumériques.

LABEL : Libellé descriptif du noeud. Format : 20 caractères alphanumériques.

MONI : Noeud de contrôle centralisé. Défaut : NOMONI.

DEV : Site de développement. Défaut : DEV.

DEV : le noeud peut être utilisé dans les espaces application et intégration.

NODEV : le noeud ne peut pas être utilisé dans les espaces application et intégration.

PROD : Site de production. Défaut : PROD.

PROD : le noeud peut être utilisé dans les espaces simulation et exploitation.

NOPROD : le noeud ne peut pas être utilisé dans les espaces simulation et exploitation.

DIR : Répertoire d'installation de la table société.

Obligatoire pour **uxadd**. Format : 60 caractères alphanumériques.

ADIR : Répertoire pour l'espace application.

Format : 60 caractères alphanumériques.

IDIR : Répertoire pour l'espace intégration.

Format : 60 caractères alphanumériques.

SDIR : Répertoire pour l'espace simulation.

Format : 60 caractères alphanumériques.

XDIR : Répertoire pour l'espace exploitation.

Format : 60 caractères alphanumériques.

DDIR : Répertoire des exécutables en développement.

Format : 60 caractères alphanumériques.

PDIR : Répertoire des exécutables en production.

Format : 60 caractères alphanumériques.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné, la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd NODE tnode=Asterix label=\"Machine de test\" dir=C:\UNIV43
xdir=C:\UNIV43\exp\data sdir=C:\UNIV43\sim\data
idir=C:\UNIV43\int\data adir=C:\UNIV43\app\data
ddir=C:\Univer$e\4.3.0f\exec pdir=C:\Univer$e\4.3.0f\exec
```

créé le noeud Asterix sur lequel la société UNIV43 est installée dans le répertoire c:\, de description "Machine de test", il est créé par défaut sur les deux univers et n'est pas noeud de contrôle centralisé.

```
uxshw NODE TNODE=Asterix
```

Affiche la définition du noeud créé précédemment.

Les types d'unités de gestion

Les commandes

uxadd MUT : crée un type d'unité de gestion dans la table des types.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MUT=	**1c**	
LABEL=	**20c**	

Exemple :

```
uxadd MUT MUT=A LABEL=Agencies
```

uxupd MUT : modifie les caractéristiques d'un type d'unité de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MUT=	**1c ***	
LABEL=	20c	

Exemple :

```
uxupd MUT MUT=S LABEL=Stocks
```

uxshw MUT : affiche les caractéristiques d'un type d'unité de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MUT=	**1c ***	
OUTPUT=file_name		

Exemple :

```
uxshw MUT MUT=A
```

uxdlt MUT : supprime un type d'unité de gestion de la table des types.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD

MUT=	**1c ***	

Exemple :

```
uxdlt MUT MUT=T
```

uxlst MUT : liste la table des types d'unité de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MUT=	1c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst MUT OUTPUT="c:\temp\LstMut.txt"
```

Description des items

MUT : Code du type de l'unité de gestion. Format : 1 caractère alphanumérique.

Saisie d'un code type (valeur générique acceptée sauf pour **uxadd**).

LABEL : Libellé descriptif du type de l'unité de gestion. Format : 20 caractères alphanumériques.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

INF et FULL sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

Exemples

```
uxadd MUT MUT=A LABEL=Agences
```

```
uxadd MUT MUT=S LABEL=Siege
```

créé les types d'unités de gestion "A" de libellé "Agences" et "S" de libellé "Siège".

```
uxlst MUT
```

Liste l'ensemble des types d'unités de gestion définis avec les libellés associés.

Les unités de gestion

Les commandes

uxadd MU : crée une unité de gestion dans la table des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c**	
TNODE=	**10c**	
LABEL=	20c	
DEV | NODEV		DEV
PROD | NOPROD		PROD

Exemple :

```
uxadd MU MU=HEADQUART LABEL=\"Company headquarters\" TNODE=%S_NOEUD%
```

uxupd MU : modifie les caractéristiques d'une unité de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=*	**10c ***	
TNODE=	10c	
LABEL=	20c	
DEV | NODEV		
PROD | NOPROD		

Exemple :

```
uxupd MU MU=%S_NOEUD% LABEL=\"Local node\"
```

uxshw MU : affiche les caractéristiques d'une unité de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c ***	
OUTPUT=file_name		

Exemple :

```
uxshw MU MU=A_TOURS
```

uxdlt MU : supprime une unité de gestion de la table des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c ***	

Exemple :

```
uxdlt MU MU=S_TOURS
```

uxlst MU: liste la table des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	10c *	*
INF | FULL | LOCAL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst MU OUTPUT="c:\temp\LstMu.txt"
```

uxdst MU : distribue la table des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY

NODE=	10c	S_NOEUD
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst MU TMU=S
```

Description des items

MU : Code de l'unité de gestion. Format : 10 caractères alphanumériques.

Saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**).

LABEL : Libellé descriptif de l'unité de gestion. Format : 20 caractères alphanumériques.

DEV : Site de développement. Défaut : DEV.

DEV : l'unité de gestion peut être utilisée dans les espaces application et intégration.

NODEV : l'unité de gestion ne peut pas être utilisée dans les espaces application et intégration.

PROD : Site de production. Défaut : PROD.

PROD : l'unité de gestion peut être utilisée dans les espaces simulation et exploitation.

NOPROD : l'unité de gestion ne peut pas être utilisée dans les espaces simulation et exploitation.

TNODE : noeud de résidence de unité de gestion. Format : 10 caractères alphanumériques.

Obligatoire avec **uxadd**.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

Si LOCAL est indiqué, seules les unités de gestion du noeud local seront listées.

Les items INF, FULL et LOCAL sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd MU mu=A_TOURS label=\"Agence de Tours\" TNODE=Asterix
uxadd MU MU=A_LENS label=\"Agence de Lens\" TNODE=%S_NOEUD%
uxadd MU MU=SIEGE label=\"Siege societe\" TNODE=%S_NOEUD%
```

créé l'unité de gestion "A_TOURS", de type "A", de libellé "Agence de Tours" définie sur le noeud distant Asterix, créé les unités de gestion A_LENS et SIEGE définies sur le noeud local.

```
uxshw MU MU=A_TOURS
```

affiche la définition de l'unité de gestion A_TOURS.

```
uxlst MU LOCAL
```

liste les unités de gestions résidant sur le noeud local, A_TOURS n'y figure donc pas.

Les dépendances des unités de gestion

Les commandes

uxadd HDP : crée une dépendance dans la table des dépendances des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c**	
DEPMU=	**10c**	

Exemple :

```
uxadd HDP MU=HEADQUART DEPMU=A_TOURS
```

uxshw HDP : affiche les caractéristiques d'une dépendance.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c ***	
DEPMU=	**10c ***	
OUTPUT=file_name		

Exemple :

```
uxshw HDP MU=HEADQUART DEPMU=*
```

uxdlt HDP : supprime une dépendance de la table des dépendances des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c ***	
DEPMU=	**10c ***	

Exemple :

```
uxdlt HDP MU=A_TOURS DEPMU=A_LENS
```

uxlst HDP : liste la table des dépendances des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	10c *	*
DEPMU=	10c *	*
OUTPUT=file_name		

Exemple :

```
uxlst HDP OUTPUT=c:\temp\LstHdp.txt
```

uxdst HDP : distribue la table des dépendances des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD

TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst HDP TMU=S
```

Description des items

MU : unité de gestion origine de la dépendance. Format : 10 caractères alphanumériques.

Saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**).

DEPMU : unité de gestion dépendante de l'unité de gestion origine. Format : 10 caractères alphanumériques.

Saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**).

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd HDP MU=SIEGE DEPMU=A_TOURS
uxadd HDP MU=SIEGE DEPMU=A_LENS
```

créé une dépendance entre l'unité de gestion SIEGE et les unités de gestion A_TOURS et A_LENS.

```
uxlst HDP MU=SIEGE
```

Affiche la liste des dépendances de l'unité de gestion SIEGE.

Les domaines

Les commandes

uxadd DOM : crée un domaine dans la table des domaines.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
DOM=	**1c**	
LABEL=	20c	" "

Exemple :

```
uxadd DOM DOM=T LABEL=\"Technical activities\"
```

uxupd DOM : modifie les caractéristiques d'un domaine.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
DOM=	**1c ***	
LABEL=	20c	

Exemple :

```
uxupd DOM DOM=V LABEL=\"General services\"
```

uxshw DOM : affichage les caractéristiques d'un domaine.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
DOM=	**1c ***	
OUTPUT=file_name		

Exemple :

```
uxshw DOM DOM=*
```

uxdlt DOM : supprime un domaine de la table des domaines.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
DOM=	**1c ***	

Exemple :

```
uxdlt DOM DOM=V
```

uxlst DOM : liste la table des domaines.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
DOM=	1c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst DOM OUTPUT="c:\temp\LstDom.txt"
```

uxdst DOM : distribue la table des domaines.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst DOM TMU=S
```

Description des items

DOM : Code du domaine. Saisie d'un code domaine (valeur générique acceptée sauf pour **uxadd**).

Format : 1 caractère alphanumérique.

LABEL : Libellé descriptif du domaine. Format : 20 caractères alphanumériques.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd DOM DOM=T LABEL=\"Activites techniques\"
```

créé un domaine T de libellé "Activités techniques".

```
uxlst DOM
```

liste les domaines définis et leurs libellés.

Les applications

Les commandes

uxadd APPL : crée une application dans la table des applications.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c**	
LABEL=	**20c**	
DOM=	**1c**	

Exemple :

```
uxadd APPL APPL=DB DOM=T LABEL=\"Data Base\"
```

uxupd APPL : modifie les caractéristiques d'une application.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c ***	
LABEL=		
DOM=	1c	

Exemple :

```
uxupd APPL APPL=BK DOM=T
```

uxshw APPL : affiche les caractéristiques d'une application.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c ***	
OUTPUT=file_name		

Exemple :

```
uxshw APPL APPL=BK
```

uxdlt APPL : supprime une application de la table des applications.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c ***	

Exemple :

```
uxdlt APPL APPL=DB
```

uxlst APPL : liste la table des applications.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	2c	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst APPL OUTPUT="c:\temp\LstAppl.txt"
```

uxdst MU : distribue la table des applications.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst APPL TMU=S
```

Description des items

APPL : Code de l'application. Format : 2 caractères alphanumériques.

Saisie d'un code application (valeur générique acceptée sauf pour **uxadd**).

LABEL : Libellé descriptif de l'application. Format : 20 caractères alphanumériques.

DOM : Code du domaine auquel l'application est associée. Format : 1 caractère alphanumérique.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

INF et FULL sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

uxadd APPL APPL=BK DOM=T LABEL=Backup

créé une application BK de libellé "Backup" appartenant au domaine T.

uxshw APPL APPL=BK

affiche la définition de l'application Backup.

Les répertoires application

Les commandes

uxadd APD : crée un répertoire application dans la table des répertoires application.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c**	
TNODE=	**10c**	
DDIR=	**60c**	
PDIR=	**60c**	

Exemple :

uxadd APD APPL=BK TNODE=Asterix DDIR="d:\temp\dev" PDIR="d:\temp\expl"

uxupd APD : modifie les caractéristiques d'un répertoire application.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c ***	
TNODE=	**10c ***	
DDIR=	60c	
PDIR=	60c	

Exemple :

uxupd APD APPL=BK TNODE=%S_NOEUD% PDIR="c:\temp\prod"

uxshw APD : affiche les caractéristiques d'un répertoire application.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c ***	
TNODE=	**10c ***	
OUTPUT=file_name		

Exemple :

uxshw APD APPL=BK TNODE=*

uxdlt APD : supprime un répertoire application de la table des répertoires application.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	**2c ***	
TNODE=	**10c ***	

Exemple :

```
uxdlt APD APPL=BK TNODE=Asterix
```

uxlst APD : liste les répertoires application.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
APPL=	2c *	*
TNODE=	10c *	*
DEV | PROD | ALL		ALL
OUTPUT=file_name		

Exemple :

```
uxlst APD OUTPUT="c:\temp\LstApd.txt"
```

uxdst APD : distribue la table des répertoires des applications.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst APD TMU=S
```

Description des items

APPL : Code de l'application. Format : 2 caractères alphanumériques.

Saisie d'un code application (valeur générique acceptée sauf pour **uxadd**).

TNODE : Code du noeud. Format : 10 caractères alphanumériques.

Saisie d'un code noeud (valeur générique acceptée sauf pour **uxadd**).

DEV : Univers de développement. Saisie incompatible avec PROD ou ALL.

Permet l'affichage des répertoires pour l'univers de développement.

PROD : Univers de production. Saisie incompatible avec DEV ou ALL.

Permet l'affichage des répertoires pour l'univers de production.

ALL : Tous les univers. Saisie incompatible avec DEV ou PROD.

Permet l'affichage des répertoires de tous les univers.

DDIR : Répertoire pour l'univers de développement.

Format : 60 caractères alphanumériques.

PDIR : Répertoire pour l'univers de production.

Format : 60 caractères alphanumériques.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd APD APPL=BK TNODE=%S_NOEUD% DDIR=c:\temp\dev PDIR=c:\temp\expl
```

créé des répertoires pour l'application BK sur le noeud local, le répertoire pour l'univers de développement est "c:\temp\dev" et pour l'univers de production est "c:\temp\expl".

```
uxshw APD APPL=BK TNODE=*
```

affiche la définition des répertoires de l'application BK.

```
uxupd APD APPL=BK TNODE=%S_NOEUD% PDIR=c:\temp\prod
```

modifie le répertoire de l'application BK sur le noeud local pour l'univers de production en "c:\temp\prod".

```
uxlst APD
```

liste les répertoires application définis.

Les répertoires unités de gestion

Les commandes

uxadd MUD : crée un répertoire unité de gestion dans la table des répertoires unité de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c**	
APPL=	**2c**	
XDIR=	**60c**	
SDIR=	**60c**	
IDIR=	**60c**	
ADIR=	**60c**	

Exemple :

```
uxadd MUD MU=HEADQUART APPL=BK XDIR="c:\temp\BK\exp"
SDIR="c:\temp\BK\sim" IDIR="c:\temp\BK\int" ADIR="c:\temp\BK\app"
```

uxupd MUD : modifie les caractéristiques d'un répertoire UG.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c ***	

APPL=	**2c ***	
XDIR=	60c	
SDIR=	60c	
IDIR=	60c	
ADIR=	60c	

Exemple :

```
uxupd MUD MU=HEADQUART APPL=BK XDIR="c:\temp\BK\prod"
```

uxshw MUD : affiche les caractéristiques d'un répertoire UG.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c ***	
APPL=	**2c ***	
APP | INT | SIM | EXP		
OUTPUT=file_name		

Exemple :

```
uxshw MUD MU=HEADQUART APPL=*
```

uxdlt MUD : supprime un répertoire UG de la table des répertoires UG. L'effacement est réalisé pour tous les espaces.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	**10c ***	
APPL=	**2c ***	

Exemple :

```
uxdlt MUD MU=HEADQUART APPL=PR
```

uxlst MUD : affiche, par espace, les répertoires des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
MU=	10c *	*
APPL=	2c *	*
APP | INT | SIM | EXP | ALL		ALL
OUTPUT=file_name		

Exemple :

```
uxlst MUD OUTPUT="c:\temp\LstMud.txt"
```

uxdst MUD : distribue la table des répertoires des unités de gestion.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
TMU=	**10c ***	

TMUT=	**1c ***	

Exemple :

```
uxdst MUD TMU=S
```

Description des items

MU : Code de L'unité de gestion. Format : 10 caractères alphanumériques.

Saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**).

APPL : Code de l'application. Format : 2 caractères alphanumériques.

Saisie d'un code application (valeur générique acceptée sauf pour **uxadd**).

APP : Espace application. Saisie incompatible avec INT, SIM, EXP ou ALL.

Permet l'affichage des répertoires pour l'espace application.

INT : Espace intégration. Saisie incompatible avec APP, SIM, EXP ou ALL.

Permet l'affichage des répertoires pour l'espace intégration.

SIM : Espace simulation. Saisie incompatible avec APP, INT, EXP ou ALL.

Permet l'affichage des répertoires pour l'espace simulation.

EXP : Espace exploitation. Saisie incompatible avec APP, INT, SIM ou ALL.

Permet l'affichage des répertoires pour l'espace exploitation.

ALL : Tous les espace. Saisie incompatible avec APP, INT, SIM ou EXP.

Permet l'affichage des répertoires pour tous les espaces.

ADIR : Répertoire pour l'espace application.

Format : 60 caractères alphanumériques.

IDIR : Répertoire pour l'espace intégration.

Format : 60 caractères alphanumériques.

SDIR : Répertoire pour l'espace simulation.

Format : 60 caractères alphanumériques.

XDIR : Répertoire pour l'espace exploitation.

Format : 60 caractères alphanumériques.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd MUD MU=SIEGE APPL=BK XDIR=c:\temp\exp SDIR=c:\temp\sim
IDIR=c:\temp\int ADIR=c:\temp\app
```

créé des répertoires pour l'unité de gestion SIEGE pour l'application BK, le répertoire pour l'espace application est "c:\temp\app", le répertoire pour l'espace intégration est "c:\temp\int", le répertoire pour l'espace simulation est "c:\temp\sim" et le répertoire pour l'espace exploitation est "c:\temp\exp".

```
uxshw MUD MU=SIEGE APPL=*
```

affiche la définition des répertoires de l'unité de gestion SIEGE pour toutes les applications.

Les profils utilisateurs

Les commandes

uxadd PROF : crée un profil utilisateur dans la table des profils.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
PROF=	**8c**	
LABEL=	40c	

Exemple :

```
uxadd PROF PROF=PROFECA LABEL=\"Test exploitation\"
```

uxupd PROF : modifie le label d'un profil utilisateur.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
PROF=	**8c ***	
LABEL=	40c	

Exemple :

```
uxupd PROF PROF=PROFECA LABEL=\"Test de l'exploitation\"
```

uxdlt PROF : supprime un profil utilisateur.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
PROF=	**8c ***	

Exemple :

```
uxdlt PROF PROF=PROFECA
```

uxlst PROF : liste les profils utilisateurs.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
PROF=	8c *	
OUTPUT=file_name		

Exemple :

```
uxlst PROF PROF=*
```

uxext AUT : extrait la définition d'un profil.

Item	Format	Défaut
SOC=	6c	S_COMPANY

NODE=	10c	S_NOEUD
AUT		
PROFIL=	**8c ***	
OUTPUT=file_name		
REPL		
NOVERIF		

Exemple :

```
uxext AUT PROFIL=* OUTPUT="c:\temp\Profil.ext" REPL
```

uxins AUT : insère la définition d'un profil dans les fichiers correspondants de Dollar Universe pour la même version de Dollar Universe.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
AUT		
PROFIL=	**8c ***	
INPUT=file_name		
REPL		

Exemple :

```
uxins AUT PROFIL=* INPUT="c:\temp\Profil.ext"
```

Description des items

PROFIL : Code du profil. Format : 8 caractères alphanumériques maximum.

Saisie d'un code profil (valeur générique acceptée sauf pour **uxadd**).

LABEL : Libellé descriptif du profil. Format : 40 caractères alphanumériques.

REPL : option de remplacement.

Indique si l'enregistrement extrait (inséré) annule et remplace l'enregistrement déjà existant dans le fichier de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signalé par un message pour chaque occurrence).

OUTPUT : stockage des résultats. Le nom du fichier est **obligatoire** pour la commande **uxext.**

Si l'item n'est pas renseigné la sortie se fait à l'écran.

INPUT : fichier d'entrée pour la commande uxins. Le nom du fichier est **obligatoire**.

NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

Exemples

```
uxadd PROF PROF=PROFECA LABEL=\"Test de l'exploitation\"
```

créé le profil PROFECA avec le libellé "Test de l'exploitation". Le caractère \ sert uniquement à préserver le guillemet.

```
uxext AUT PROFIL=* OUTPUT="c:\temp\extAutProf.txt" REPL
```

extrait la définition de tous les profils et les écrit dans le fichier résultat c:\temp\extAutProf.txt. Si certains enregistrements existent déjà dans ce fichier, il seront écrasés.

Les utilisateurs

Les commandes

uxadd USER : crée un utilisateur dans la table des utilisateurs.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
USER=	**12c**	
CODE=	**3c**	
PROF=	**8c**	
DEV | PROD | BOTH		

Exemple :

```
uxadd USER USER=admin CODE=101 PROF=PROFEXP BOTH
```

uxupd USER : modifie les caractéristiques d'un utilisateur.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
USER=	**12c ***	
PROF=	8c	

Exemple :

```
uxupd USER USER=admin PROF=PROFADM
```

uxshw USER : affiche les caractéristiques d'un utilisateur.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
USER=	**12c ***	
CODE=	3n *	
DEV | PROD | BOTH		
OUTPUT=file_name		

Exemple :

```
uxshw USER USER=admin
```

uxdlt USER : supprime un utilisateur de la table des utilisateurs.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
USER=	**12c ***	

Exemple :

```
uxdlt USER USER=eca
```

uxlst USER : liste la table des utilisateurs.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
USER=	12c *	*
CODE=	3n *	
DEV | PROD | BOTH		
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst USER OUTPUT="c:\temp\LstUser.txt"
```

uxdst USER : distribue les utilisateurs.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
USER=	**12c ***	
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst USER USER=* TMU=S
```

Description des items

USER : Code utilisateur : correspond au nom indiqué lors du login. Format : 12 caractères alphanumériques.

Saisie d'un code utilisateur (valeur générique acceptée sauf pour **uxadd**).

CODE : Code auteur associé à l'utilisateur. Format : 3 caractères numériques.

PROF : Profil associé à l'utilisateur. Format : 8 caractères alphanumériques.

DEV : Type du compte de soumission.

Pour la commande **uxadd**, une des trois valeurs suivantes est obligatoire. Il n'y a pas de valeur par défaut.

- DEV : l'utilisateur ne peut être utilisé comme compte de soumission que dans les espaces application et intégration.
- PROD : l'utilisateur ne peut être utilisé comme compte de soumission que dans les espaces simulation et exploitation.
- BOTH : l'utilisateur peut être utilisé comme compte de soumission dans tous les espaces.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd USER USER=admin CODE=101 PROF=PROFADM BOTH
```

créé l'utilisateur admin avec le code auteur 101, un profil PROFADM et autorisé à être utilisé comme compte de soumission dans tous les espaces.

```
uxshw USER USER=admin
```

affiche la définition de l'utilisateur admin.

Définition des uprocs et sessions

Les classes

Les commandes

uxadd CLASS : crée une classe dans la table des classes.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
CLASS=	**6c**	
LABEL=	20c	" "

Exemple :

```
uxadd CLASS CLASS=UPDATE LABEL=\"General updates\"
```

uxupd CLASS : modifie les caractéristiques d'une classe.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
CLASS=	**6c ***	
LABEL=	20c	

Exemple :

```
uxupd CLASS CLASS=EDIT LABEL=\"General printouts\"
```

uxshw CLASS : affiche les caractéristiques d'une classe.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
CLASS=	**6c ***	
OUTPUT=file_name		

Exemple :

```
uxshw CLASS CLASS=EDIT
```

uxdlt CLASS : supprime une classe de la table des classes.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD

CLASS=	**6c ***	

Exemple :

```
uxdlt CLASS CLASS=UPDATE
```

uxlst CLASS : liste la table des classes.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
CLASS=	6c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst CLASS OUTPUT="c:\temp\LstClass.txt"
```

uxdst CLASS : distribue les classes.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
CLASS=	**6c ***	
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst CLASS CLASS=* TMU=S
```

Description des items

CLASS : Code de la classe. Saisie d'un code classe (valeur générique acceptée sauf pour **uxadd**).

Format : 6 caractères alphanumériques.

LABEL : Libellé descriptif de la classe. Format : 20 caractères alphanumériques.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd CLASS CLASS=EDIT LABEL=Editions
uxadd CLASS CLASS=UPDATE LABEL=\"Mises a jour\"
```

créé les classes EDIT de libellé "Editions" et UPDATE de libelle "Mises à jour".

```
uxlst CLASS
```

liste les classes définies.

Les ressources

Les commandes

uxadd RES : crée une ressource dans la liste des ressources.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c**	
LABEL=	40c	" "
NATURE=FIL | LOG | DSK | PRC | QUE | SYS	**3c**	
FNAME=file_name	*	
DIR=dir_name	*	
LOCK | UNLOCK		UNLOCK
CLEAR | NOCLEAR		CLEAR
QT1=	4n	0
QT2=	4n	0
FREQ=	30...86400	120

Exemples :

```
uxadd RES RES=TESTFILE LABEL=\"Awaiting test file\" NATURE=FIL
DIR=c:\\temp\\ FNAME=TestFile FREQ=30
```

```
uxadd RES RES=D_BASE LABEL=\"Base protection\" NATURE=LOG
FNAME=!UG!!ESP! QT1=10 QT2=10 NOCLEAR
```

uxupd RES : modifie les caractéristiques d'une ressource.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	
LABEL=	40c	
NATURE=FIL | LOG | DSK | PRC | QUE | SYS	3c	
FNAME=file_name	*	
DIR=dir_name	*	
LOCK | UNLOCK		
CLEAR | NOCLEAR		
QT1=	4n	
QT2=	4n	
FREQ=	30...86400	

Exemple :

```
uxupd RES RES=D_FIL_PAY FNAME=pay_!DTRAIT!
```

uxshw RES : affiche la définition et l'allocation d'une ressource.

Item	Format	Défaut
SOC=	6c	S_COMPANY

NODE=	10c	S_NOEUD
RES=	**20c ***	
ESP=X | S | I | A | *	1c * " "	*
MU=	10c * " "	*
TSOC=	6c * " "	*
PDATE=	YYYYMMDD * " "	*
INF | RSV | FULL		INF
OUTPUT=file_name		

Exemple :

```
uxshw RES RES=D_*
```

uxdlt RES : supprime une ressource.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	

Exemple :

```
uxdlt RES RES=TESTFILE
```

uxlst RES : liste les ressources.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	20c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst RES OUTPUT="c:\temp\LstRes.txt"
```

uxdst RES : distribue les ressources.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
RES=	**20c ***	
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst RES RES=* TMU=S
```

uxext RES : extrait la définition complète d'une ressource.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=*	**20c ***	

REPL **OUTPUT=file_name** NOVERIF		

Exemple :

```
uxext RES RES=D_* OUTPUT="c:\temp\Res.ext"
```

uxins RES : insère la définition complète d'une ressource dans les fichiers correspondants de Dollar Universe pour la même version de Dollar Universe.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=*	**20c ***	
REPL **INPUT=file_name**		

Exemple :

```
uxins RES RES=* INPUT="c:\temp\Res.ext"
```

Description des items

RES : référence de la ressource. Format : 20 caractères alphanumériques.

Requiert la saisie de la référence de la ressource (valeur générique acceptée sauf pour uxadd).

LABEL : libellé descriptif de la ressource. Format : 40 caractères alphanumériques.

NATURE : définit la nature de la ressource.

Sur VMS, les natures possibles sont FIL (fichier), LOG (logique) DSK (disques), PRC (process), LNM (noms logiques), QUE (queues batch), SYS (système).

Sur UNIX et Windows, les valeurs possibles sont FIL (fichier) et LOG (logique).

FNAME : nom du fichier ou de la ressource.

Permet la saisie du nom du fichier associé à une ressource de type FIL ou, dans le cas d'une ressource logique, permet la saisie d'un nom de "fichier virtuel" sur lequel repose la vérification de la ressource.

Ce nom peut contenir les variables !UG!, !DTRAIT!, !ESP et !SOC! qui seront analysées lors du pilotage de l'uproc contenant une condition sur cette ressource (en fonction de son contexte d'exécution). Il peut contenir également le caractère générique "*".

FREQ : cycle de vérification par le surveillant.

Durée de la période de vérification : de 30 à 86400 secondes. La valeur par défaut est 120s.

DIR : localisation du fichier.

Permet la saisie, pour une ressource de type FIL uniquement, du chemin d'accès au fichier.

Ce nom peut contenir les variables !UG!, !DTRAIT!, !ESP et !SOC! qui seront analysées lors du pilotage de l'uproc contenant une condition sur cette ressource (en fonction de son contexte d'exécution). Il peut contenir également le caractère générique "*".

UNLOCK : allocation de la ressource.

Indique que l'uproc conditionnée pourra s'attribuer tout ou partie des quotas de cette ressource.

Si le mot clé LOCK est utilisé, l'uproc conditionnée ne pourra s'attribuer que la totalité de la ressource durant son exécution et aucune autre uproc, conditionnée par cette ressource ne pourra s'exécuter.

Usage de LOCK incompatible avec QT1 ou QT2.

QT1 ou **QT2** : quotas d'allocation de niveau 1 et 2 respectivement.

Dans les commandes **uxadd** ou **uxupd**, ils permettent de définir les valeurs maximum de quotas qui pourront être alloués lors de l'exécution d'une commande **uxhld** ou lors de l'exécution d'une uproc portant une condition sur cette ressource. L'utilisation de QT1 et QT2 est incompatible avec LOCK.

Format : de 0 à 9999.

CLEAR : libération. Permet de libérer la ressource sélectionnée,.

Dans les commandes **uxadd** ou **uxupd**, la libération ne sera effective que lors de la terminaison de l'uproc conditionnée. L'uproc conditionnée ne libère que les quotas qu'elle a alloués.

NOCLEAR indique que la ressource ne sera pas libérée en fin d'exécution de l'uproc conditionnée, la commande **uxrls** devra être exécutée pour permettre la libération de la ressource.

INF : informations techniques. Permet l'affichage des informations de définition de la ressource.

Autres valeurs : FULL et RSV : cf. chapitre "Les commandes d'exploitation" paragraphe "Les ressources" page 83.

Les trois items sont incompatibles entre eux.

REPL : option de remplacement.

Indique si l'enregistrement extrait (inséré) annule et remplace l'enregistrement déjà existant dans le fichier de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signalé par un message pour chaque occurrence).

OUTPUT : stockage des résultats. Le nom du fichier est **obligatoire** pour la commande **uxext.**

Si l'item n'est pas renseigné la sortie se fait à l'écran.

INPUT : fichier d'entrée pour la commande uxins. Le nom du fichier est **obligatoire**.

NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

Exemples

```
uxadd RES RES=D_FIL_PAY LABEL=\"Attente fichier paye\" nature=FIL
FNAME=paye_!DTRAIT! DIR=c:\temp\ LOCK CLEAR
```

```
uxadd RES RES=D_BASE LABEL=\"Protection base\" nature=LOG
FNAME=!UG!!ESP! DIR=c:\temp QT1=10 QT2=10 NOCLEAR
```

crée deux ressources :

- la ressource D_FIL_PAY portant sur l'arrivée du fichier c:\temp\paye_!DTRAIT!, la ressource est exclusive et la libération est automatique en fin d'exécution de l'uproc conditionnée,
- la ressource D_BASE portant sur une protection d'accès à une base, la ressource est logique, elle est spécifique à l'espace et à l'unité de gestion d'exécution de l'uproc conditionnée, elle peut être partagée et devra être libérée par une commande **uxrls** (cf. chapitre "Les commandes d'exploitation" paragraphe "Les ressources" page 83) dans le script d'une uproc.

```
uxshw RES RES=D_*
```

affiche la définition des ressources créées précédemment.

```
uxext RES RES=D_* output=c:\temp\DOC.ext
```

extrait toutes les ressources commençant par D_ de l'environnement courant et archive les données dans le fichier c:\temp\DOC.ext.

```
uxins RES RES=* input=c:\temp\DOC.ext
```

insère toutes les ressources du fichier c:\temp\DOC.ext dans l'environnement courant (société, noeud).

Les uprocs

Les commandes

uxadd UPR : crée une uproc dans la liste des uprocs : seules les informations générales sont définies. Les informations complémentaires sont saisies par l'intermédiaire d'une commande de mise à jour de l'uproc.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c**	
VUPR=	3n	

GENINF : informations générales

Item	Format	Défaut
LABEL=	60c	" "
FPERIOD=N | D | W | T | F | M | 2 | 3 | Q | S | Y	1c	N
CLFIL=file_name		
DOM=	1c	
APPL=	**2c**	
CLASS=	6c	
MEMO=N | O | A	1c	N
NBPER=	2n	0

Exemple :

```
uxadd UPR EXP UPR=D_LOAD_FIL LABEL=\"Loading of file\" GENINF
FPERIOD=W APPL=BK MEMO=O NBPER=1 CLFIL="c:\temp\D_LOAD_FIL.bat"
```

uxupd UPR : modifie les informations générales d'une uproc, ajoute des caractéristiques complémentaires telles que des conditions, des consignes, des incompatibilités.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c ***	
VUPR=	3n *	

Une seule des familles décrites ci-dessous peut être utilisée dans une commande uxupd.

GENINF : informations générales

Item	Format	Défaut
LABEL=	60c	
FPERIOD=N | D | W | T | F | M | 2 | 3 | Q | S | Y	1c	
CLFIL=file_name		
DOM=	1c	
APPL=	2c	
CLASS=	6c	
MEMO=N | O | A	1c	
NBPER=	2n	

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL GENINF FPERIOD=D
```

LAUNCHVAR add : ajoute une variable.

Item	Format	Format	Format
VARNAME=	**20c**	**20c**	**20c**
VTYPE=D | Q | T	D	Q	T (défaut)
VFORMAT=	**"YYYY/MM/DD" "MM/DD/YYYY" "DD/MM/YYYY"**		3n
VALMIN=		12n	
VALMAX=		12n	
VVALUE=	**"10c"**	**12n**	**255c**

Exemples :

```
uxupd UPR EXP UPR=D_LOAD_FIL LAUNCHVAR add VARNAME=VARDATE VTYPE=D
VFORMAT="YYYY/MM/DD" VVALUE="2002/07/01"
```

```
uxupd UPR EXP UPR=D_LOAD_FIL LAUNCHVAR add VARNAME=VARNUM VTYPE=Q
VVALUE=123
```

```
uxupd UPR EXP UPR=D_LOAD_FIL LAUNCHVAR add VARNAME=VARTXT
VVALUE=\"Last var\"
```

LAUNCHVAR dlt : supprime une variable.

Item	Format	Défaut
VARNAME=	**20c**	

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL LAUNCHVAR dlt VARNAME=VARTXT
```

INCCLASS add : ajoute une classe d'incompatibilité.

Item	Format	Défaut
XCLASS=	**6c**	

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL INCCLASS add XCLASS=BACKUP
```

INCCLASS dlt : supprime une classe d'incompatibilité.

Item	Format	Défaut
XCLASS=	**6c**	

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL INCCLASS dlt XCLASS=BACKUP
```

TERMINS add : ajoute une consigne de terminaison.

Item	Format	Défaut
TERMNO=	**2n**	
TERM=	**10c**	
PDATE=	ANY | SAME | (±n,n,n) | (a, ,n)	SAME
MU=	SAME | "mu(MU_CODE)" | "mut(T)" | "hdp({XY})"	SAME
SESS=	any | same | samerun | expl(SESS_CODE)	ANY
USER=	any | same	ANY
STATUS=	terminated | abnormal	TERMINATED

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL TERMINS add TERMNO=01 TERM=D_LOAD_FIL
PDATE="(-1,,)" STATUS=ABNORMAL
```

TERMINS dlt : supprime une consigne de terminaison.

Item	Format	Défaut
TERMNO=	**2n**	

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL TERMINS dlt TERMNO=01
```

DEPCON add : ajoute une condition d'enchaînement.

Item	Format	Défaut
CONDNO=	**2n**	
DEP=	**10c**	
PDATE=	ANY | SAME | (±n,n,n) | (a, ,n)	SAME
DTYPE=	A | W | C | H	
MU=	SAME | "mu(MU_CODE)" | "mut(T)" | "hdp({XY})"	SAME
ONE | ALL		ONE
SESS=	any | same | samerun | expl(SESS_CODE)	ANY
USER=	ANY | SAME	ANY
STATUS=	TERMINATED | ABNORMAL | ABSENT	TERMINATED
EXPECT | EXCLUD		EXPECT
NOFATAL | FATAL		NOFATAL

Exemples :

```
uxupd UPR EXP UPR=D_BACKUP DEPCON add CONDNO=01 DEP=D_LOAD_FIL
MU="mu(SATURN)"
```

```
uxupd UPR EXP UPR=D_BACKUP DEPCON add CONDNO=02 DEP=D_LOAD_FIL
MU="hdp({AH})"
```

DEPCON dlt : supprime une condition d'enchaînement.

Item	Format	Défaut
CONDNO=	**2n**	

Exemple :

```
uxupd UPR EXP UPR=D_BACKUP DEPCON dlt CONDNO=01
```

INCCON add : ajoute une condition de non simultanéité.

Item	Format	Défaut
CONDNO=	**2n**	
INC=	**10c**	
PDATE=	SAME | ANY	SAME
MU=	SAME | "mu(MU_CODE)" | "mut(T)" | "hdp({XY})"	SAME
SESS=	any | same | samerun | expl(SESS_CODE)	ANY
USER=	ANY | SAME	ANY
EXPECT | EXCLUD		EXPECT
NOFATAL | FATAL		NOFATAL

Exemple :

```
uxupd UPR EXP UPR=D_BACKUP INCCON add CONDNO=01 INC=D_BACKUP PDATE=ANY
```

INCCON dlt : supprime une condition de non simultanéité.

Item	Format	Défaut
CONDNO=	**2n**	

Exemple :

```
uxupd UPR EXP UPR=D_BACKUP INCCON dlt CONDNO=02
```

RESCON add : ajoute une condition de ressource.

Item	Format	Défaut
CONDNO=	**2n**	
RES=	**10c**	
PDATE=	ANY | SAME | (±n,n,n) | (a, ,n)	SAME
DTYPE=	A | W | C | H	
MU=	SAME | "mu(MU_CODE)" | "mut(T)" | "hdp({XY})"	SAME
ONE | ALL		ONE
VERIF | NOVERIF		VERIF
UNLOCK | LOCK		UNLOCK
EXPECT | EXCLUD		EXPECT
NOFATAL | FATAL		NOFATAL
ATTRIB=		

OPER=	# | = | < | > | <= | >=	
VALUE=	255c	
QT1=	4n	0
QT2=	4n	0

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL RESCON add CONDNO=01 RES=D_FIL_PAY
```

RESCON dlt : supprime une condition de ressource.

Item	Format	Défaut
CONDNO=	**2n**	

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL RESCON dlt CONDNO=03
```

FORMULA : définit la formule de lancement.

Item	Format	Défaut
LFORM=	**255c**	

Exemple :

```
uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM=\"=C01 AND =C02\"
```

uxdup UPR : duplique une uproc ainsi que l'ensemble de ses caractéristiques. Si cette uproc a un script interne, le script est également dupliqué.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c**	
VUPR=	**3n**	
TUPR=	**10c**	
TVUPR=	**3n**	
TLABEL	60c	

Exemple :

```
uxdup UPR EXP UPR=IU_TEST VUPR=000 TUPR=IU_TEST2 TVUPR=000
TLABEL=\"Uproc de test\"
```

uxshw UPR : affiche les caractéristiques d'une uproc, son script (s'il est interne), sa formule de lancement et le détail des conditions. Pour ne pas afficher le script (interne) se reporter au manuel d'administration.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c** *	
VUPR=	3n *	
DOM=	1c *	
APPL=	2c *	

SCRIPT OUTPUT=file_name		

Exemple :

```
uxshw UPR EXP UPR=D_LOAD_FIL
```

uxdlt UPR : supprime une uproc.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c ***	
VUPR=	3n *	

Exemple :

```
uxdlt UPR UPR=TEST*
```

uxlst UPR : liste les uprocs.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	10c *	*
VUPR=	3n *	*
DOM=	1c *	*
APPL=	2c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst UPR OUTPUT="c:\temp\lstUpr.txt"
```

uxtra UPR : transfert les uprocs dans un autre espace de la même société sur le même noeud.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
APP | INT | SIM	3c	S_ESPEXE
UPR=	**10c ***	
VUPR	3n *	
TESP=	**I | S | X**	

Exemple :

```
uxtra UPR APP UPR=D_* TESP=X
```

uxdst UPR : distribue les uprocs.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
EXP | SIM | INT | APP	3c	S_ESPEXE

UPR=	**10c ***	
VUPR	3n *	
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst UPR UPR=* TMU=S
```

uxscv UPR : permet de définir la version courante d'une uproc (espace application et intégration uniquement).

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
INT | APP	3c	S_ESPEXE
UPR=	**10c ***	
VUPR=	3n *	

Exemple :

```
uxscv UPR APP UPR=D_* VUPR=002
```

uxrls UPR : permet de libérer une uproc.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c ***	
VUPR=	3n *	

Exemple :

```
uxrls UPR UPR=*
```

uxext UPR : extrait la définition complète d'une uproc (avec son CL si celui-ci est déclaré interne).

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c ***	
VUPR=	3n *	
REPL		
ENABLE | NOENABLE		ENABLE
OUTPUT=file_name		
NOVERIF		

Exemple :

```
uxext UPR EXP UPR=D_* OUTPUT="c:\temp\upr.ext"
```

uxins UPR : insère la définition complète d'une uproc dans les fichiers correspondants de Dollar Universe (avec son éventuel CL) pour le même espace et la même version de Dollar Universe que l'extraction.

Item	Format	Défaut
SOC=	6c	S_COMPANY

NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c ***	
VUPR=	3n *	
REPL		
ENABLE | NOENABLE		ENABLE
INPUT=file_name		

Exemple :

```
uxins UPR EXP UPR=* INPUT=c:\temp\upr.ext
```

Description des familles

Une seule des familles décrites ci-dessous peut être utilisée dans une commande **uxupd**.

GENINF : informations générales.

Permet la mise à jour des informations générales de l'uproc.

LAUNCHVAR : Variables d'une uproc.

Permet la création et la suppression des variables de l'uproc et leur valorisation.

INCCLAS : classes d'incompatibilité.

Permet de définir des classes d'incompatibilité pour l'uproc. La classe d'appartenance de l'uproc est définie dans les informations générales.

TERMINS : consignes de terminaison.

Permet de définir des consignes de terminaisons qui seront effectuées lors de la terminaison de l'uproc.

DEPCON : condition d'enchaînement.

Permet de définir une condition d'enchaînement à l'uproc ainsi que les informations de détail associées.

RESCON : condition de ressource.

Permet de définir une condition de ressource à l'uproc ainsi que les informations de détail associées.

INCCON : condition de non simultanéité.

Permet de définir une condition de non simultanéité à l'uproc ainsi que les informations de détail associées.

FORMULA : formule de lancement.

Permet la mise à jour de la formule de lancement. Cette famille doit être utilisée dès lors qu'une condition a été ajoutée ou supprimée à l'uproc pour que les conditions soient effectivement examinées lors du pilotage.

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée sauf pour **uxadd**).

TUPR : Code de l'uproc résultat de la duplication. Format : 10 caractères alphanumériques.

VUPR : Version d'uproc. Format : 3 caractères numériques.

Permet la saisie d'une version d'uproc (valeur explicite).

Par défaut, pour la commande **uxext**, toutes les versions des uprocs renseignées sont extraites (inutile avec SIM et EXP).

Pour les autres commandes la version est obligatoire dans les espaces APP et INT, elle est facultative dans les espaces SIM et EXP (systématiquement à 000).

TVUPR : Version de l'uproc résultat de la duplication. Format : 3 caractères numériques.

DOM : Code domaine. Format : 1 caractère alphanumérique.

Permet la sélection des uprocs rattachées à un domaine.

APPL : Code application. Format : 2 caractères alphanumériques.

Permet la sélection des uprocs rattachées à une application (valeur générique acceptée).

LABEL : Libellé descriptif de l'uproc. Format : 60 caractères alphanumériques.

TLABEL : Libellé descriptif de l'uproc résultat de la duplication. Format : 60 caractères alphanumériques.

FPERIOD : Période fonctionnelle.

Indique le format de la date de traitement qui sera associée à l'exécution de l'uproc.

Valeurs possibles :

N : sans	F : quinzaine	Q : quadrimestre
D : jour	M : mois	S : semestre
W : semaine	2 : bimestre	Y : année
T : décade	3 : trimestre	

CLFIL : CL associé à l'uproc. Format : 62 caractères.

Indique le chemin d'accès et le nom de la procédure associée à l'uproc.

DOM : Code domaine. Format : 1 caractère alphanumérique.

Indique à quel domaine l'uproc est rattachée.

APPL : Code application. Format : 2 caractères alphanumériques.

Indique à quelle application l'uproc est rattachée.

CLASS : Classe d'appartenance de l'uproc. Format : 6 caractères alphanumériques.

Indique à quelle classe l'uproc est associée.

MEMO : Mémorisation de l'uproc. La valeur par défaut est : **N**.

Indique le mode de mémorisation de l'uproc. Trois valeurs sont possibles :

- N : sans mémorisation,
- O : une exécution,
- A : toutes les exécutions.

Dans le cas où l'uproc est mémorisée (choix O ou A), cette information doit être complétée par l'item NBPER.

NBPER : Nombre de périodes de mémorisation de l'uproc. Format : 2 caractères numériques.

La valeur par défaut est : 0 (sans limitation).

Indique le nombre de périodes (fonctionnelles) sur lesquelles les événements seront conservés dans la base des événements.

REPL : Option de remplacement.

Indique si l'enregistrement extrait (inséré) annule et remplace l'enregistrement déjà existant dans le fichier de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signalé par un message pour chaque occurrence).

ENABLE : Option de libération.

Indique si l'uproc sera insérée dans l'état libéré. L'action de libération sera nécessaire avant toute modification d'uproc si l'objet est inséré avec le qualifier NOENABLE.

Défaut : ENABLE.

VARNAME : Nom de la variable. Format : 20 caractères alphanumériques.

Requiert la saisie du nom de la variable de l'uproc, saisie obligatoire.

VTYPE : Type de la variable. Format : Q pour quantité, T pour texte ou D pour date (choix exclusif).

La variable peut être de type texte, quantité ou date. Du type dépend le contrôle du format de la variable. Saisie obligatoire.

VFORMAT : Format de la variable.

Pour une variable de type Date, le format doit être choisi entre les trois valeurs suivantes : AAAA/MM/JJ ou JJ/MM/AAAA ou MM/JJ/AAAA.

Pour une variable de type Texte, le format exprime la longueur maximum du texte saisi entre guillemets (non compris) : au maximum et par défaut 255 caractères.

Saisie obligatoire sauf pour une variable de type Quantité pour laquelle cette information ne présente pas de sens.

VALMIN, **VALMAX** : Plage pour une variable de type Quantité uniquement. Saisie non obligatoire.

VALMIN et VALMAX doivent indiquer dans quelle plage la valeur de la variable devra être saisie. VALMIN indique la valeur inférieure, VALMAX la valeur supérieure de cette plage.

Format : 12 caractères numériques.

Valeurs par défaut : 0 et 999999999999

VVALUE : Valeur de la variable. Saisie obligatoire.

La valeur de la variable doit être compatible avec son type, son format et éventuellement sa plage, dans le cas contraire la commande sera refusée.

Si la valeur contient des caractères non alphanumériques (espaces, /…) elle doit être encadrée de guillemets.

XCLASS : Classe d'incompatibilité de l'uproc.

Indique que l'uproc est incompatible avec une uproc de cette classe.

Format : entre parenthèses, les différentes classes d'incompatibilité sont séparées par des virgules, les espaces sont interdits.

Exemple : XCLASS=(CLASSE1,CLASSE2) : l'uproc est déclarée incompatible avec les classes CLASSE1 et CLASSE2.

TERMNO : Numéro désignant la consigne de terminaison. Format : 2 caractères numériques.

Un numéro est associé à chaque consigne, deux consignes différentes ne peuvent pas porter le même numéro.

TERM : Consigne de terminaison. Format : 10 caractères alphanumériques.

Indique le code de l'uproc correspondant à l'événement qui doit être purgé par l'exécution de la consigne de terminaison.

CONDNO : Numéro désignant la condition. Format : 2 caractères numériques.

Un numéro est associé à chaque condition (quel que soit le type), deux conditions différentes ne peuvent pas porter le même numéro.

DEP : Condition d'enchaînement. Format : 10 caractères alphanumériques.

Indique le code de l'uproc correspondant à la condition d'enchaînement.

INC : Condition de non simultanéité. Format : 10 caractères alphanumériques.

Indique le code de l'uproc correspondant à la condition de non simultanéité.

RES : Condition de ressource. Format : 20 caractères alphanumériques.

Indique la référence de la ressource correspondant à la condition de ressource.

PDATE : Contrôle sur la date de traitement.

Indique le décalage recherché entre la date de traitement de l'uproc conditionnée et de l'uproc ou de la ressource conditionnante.

Format : il dépend de la période fonctionnelle de l'uproc conditionnante, il s'exprime en "+ ou -" un nombre d'unités calendaire (cf. ci-dessous) entre parenthèses, séparés par des virgules.

Si une des unités calendaire contrôlée doit être la même, aucune valeur n'est requise. Par exemple PDATE=(+1,-1,) pour une période fonctionnelle jour peut être traduit par : jour +1, mois –1, même année.

Si le contrôle doit porter sur la même date de traitement (globalement), le mot clé SAME peut être utilisé, si le contrôle doit porter sur la même unité calendaire un caractère blanc peut être utilisé dans le triplet (cf. Exemple ci-dessus),

Si le contrôle ne tient pas compte de la date de traitement le mot clé ANY peut être utilisé, si le contrôle ne doit pas tenir compte d'une unité calendaire particulière la lettre a peut être utilisée dans le triplet, par exemple PDATE=(a, ,).peut être traduit par : quel que soit le jour pour le même mois et la même année. Le tableau suivant résume les unités calendaire proposées en fonction de la période fonctionnelle de l'uproc :

	Unités calendaires									
Période fonctionnelle	jour	semaine	décade	quin-zaine	mois	2 mois	3 mois	4 mois	6 mois	année
Sans										
Jour	*				*					*
Semaine		*								*
Décade			*		*					*
Quinzaine				*	*					*
Mois					*					*
Bimestre						*				*
Trimestre							*			*
Quadrimestre								*		*
Semestre									*	*
Année										*

DTYPE: Contrôle sur la date de traitement.

Indique le type de jour (indiqué par PDATE) à considérer pour les uprocs à période fonctionnelle jour.

Format:

- A: (any) quelque que soit le type de jour : ouvré, chômé, férié : le nombre indiqué par PDATE est calculé en jours calendaires,
- W: (worked) le nombre indiqué par PDATE est calculé en jours ouvrés (en relation avec le calendrier de référence de l'unité de gestion considérée),
- C: (closed) le nombre indiqué par PDATE est calculé en jours chômés,
- H: (holiday) le nombre indiqué par PDATE est calculé en jours fériés.

MU : Contrôle sur les unités de gestion.

Indique sur quel ensemble d'unités de gestion la condition doit être recherchée.

Si le contrôle doit porter sur la même unité de gestion, le mot clé **SAME** peut être utilisé.

Si le contrôle doit porter sur une unité de gestion spécifique, l'expression du contrôle sera **mu(MU_CODE)**, où MU_CODE représente le code de l'unité de gestion visée (10 caractères alphanumériques).

Si le contrôle doit porter sur un type d'unité de gestion, l'expression du contrôle sera **mut(T)**, où T représente le type d'unité de gestion (1 caractère alphanumérique).

Si le contrôle doit porter sur des dépendances d'unité de gestion, l'expression du contrôle sera **hdp({XY})**, où X représente le type des unités de gestion visées (1 caractère alphanumérique) et Y le type d'unité de gestion origine de la dépendance (1 caractère alphanumérique).

ONE : Unités de gestion visées. La valeur par défaut est ONE.

Utilisé uniquement pour les conditions d'enchaînement ou de ressource, dans le cas où le contrôle des unités de gestion porte sur un type d'unité de gestion ou un TIH.

Le choix ONE signifie que la condition est attendue sur au moins une des unités de gestion visées, le choix ALL signifie que la condition est attendue sur toutes les unités de gestion visées.

SESS : Contrôle sur la session. Indique sur quelle session la condition doit être recherchée.

Si le contrôle ne prend pas en compte la notion de session, le mot clé **any** peut être utilisé.

Si le contrôle porte sur la même session, le mot clé **same** peut être utilisé.

Si le contrôle porte sur la même session et la même exécution, le mot clé **samerun** peut être utilisé.

Si le contrôle porte sur une session spécifique, l'expression du contrôle sera **expl(COD_SESS)**, où COD_SESS est le code de la session visée (10 caractères alphanumériques).

STATUS : Etat recherché.

Indique quel est état recherché de l'uproc (condition d'enchaînement ou consigne de terminaison uniquement).

Si l'état recherché est "TERMINE" le mot clé TERMINATED peut être utilisé (valeur par défaut).

Si l'état recherché est "INCIDENTE" le mot clé ABNORMAL peut être utilisé.

Si l'état recherché est "ABSENT" le mot clé ABSENT peut être utilisé.

USER : Compte de soumission.

Indique avec quel type de contrôle sur le compte de soumission la condition doit être recherchée.

Si le contrôle porte sur le même compte utilisateur le mot clé SAME peut être utilisé.

Si le contrôle ne prend pas en compte la notion de compte de soumission le mot clé ANY peut être utilisé (valeur par défaut).

EXPECT : logique booléenne.

Indique que la condition (ou la consigne) exprimée est recherchée vraie (valeur par défaut).

Si la condition ou la consigne recherchée doit être le contraire de ce qui est exprimé, le mot clé EXCLUD peut être utilisé.

NOFATAL : comportement du lanceur vis à vis de l'absence de la condition.

Si la condition exprimée n'est pas vérifiée lors du premier pilotage de l'uproc, le Lanceur placera le lancement en attente d'événement. L'uproc sera à nouveau pilotée lors de l'arrivée de l'événement attendu (valeur par défaut).

Si la condition exprimée n'est pas vérifiée lors du premier pilotage de l'uproc et que le mot clé FATAL est utilisé, le pilotage sera abandonné par le Lanceur et le lancement sera placé dans l'état "refusé au pilotage".

VERIF : vérification du fichier.

Indique si l'automate doit vérifier la présence physique de la ressource lors du pilotage de l'uproc portant la condition (valeur par défaut).

Si l'automate ne doit pas vérifier la présence physique de la ressource (elle ne sera vue alors que comme une ressource logique) le mot clé NOVERIF peut être utilisé.

UNLOCK : allocation de la ressource.

Indique que l'uproc conditionnée ne s'attribue que les quotas définis dans la condition (valeur par défaut). D'autres uprocs conditionnées sur cette ressource pourront s'exécuter dans la mesure des quotas disponibles.

Si le mot clé LOCK est utilisé, l'uproc conditionnée s'attribue la totalité de la ressource durant son exécution et aucune autre uproc, conditionnée par cette ressource ne pourra s'exécuter.

Si la condition de ressource est déclarée UNLOCK mais que la ressource a été définie "LOCK" le comportement sera identique.

ATTRIB : attribut à vérifier. Permet la saisie de l'attribut de la nature de la ressource.

Pour la nature FIL, les attributs EXIST et SIZE sont proposés.

OPER : opérateur. Utilisé pour les ressources de nature FIL avec l'attribut SIZE.

VALUE : valeur. Utilisé pour les ressources de nature FIL avec l'attribut SIZE.

QT1 et **QT2** : quotas nécessaires. Valeur numérique comprise entre 0 et 9999, 0 par défaut.

Dans le cas d'une condition et d'une ressource déclarées UNLOCK, le Lanceur examine, au moment du pilotage, si les quotas exprimés dans la condition sont effectivement disponibles. S'ils le sont, l'uproc sera soumise à l'exécution et bloquera les quotas correspondants. Plusieurs uprocs peuvent ainsi se partager une même ressource.

LFORM : formule de lancement.

Exprime la totalité de la formule de lancement pour l'uproc visée et l'action en cours. Sa longueur maximum doit être de 400 caractères.

L'expression est la même qu'en interface caractère :

Cnn : indique la condition numéro nn,

= : indique que la condition est recherchée vraie,

: indique que la condition est recherchée fausse,

AND ou **ET** : indique que les deux conditions doivent être vérifiées,

OR ou **OU** : indique que l'une ou l'autre des deux conditions doivent être vérifiées.

le "AND" a priorité sur le "OR", des parenthèses peuvent être utilisées pour définir des groupes de conditions réunies par des "OR".

Par exemple : LFORM="((=C01 AND (=C02 OR #C03)))"

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

INF et FULL sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Le nom du fichier est **obligatoire** pour la commande **uxext.**

Si l'item n'est pas renseigné la sortie se fait à l'écran.

INPUT : fichier d'entrée pour la commande uxins. Le nom du fichier est **obligatoire**.

TESP : code de l'espace cible du transfert : X pour Exploitation, S pour Simulation ou I pour intégration.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

SCRIPT : peut être utilisé avec la commande **uxshw upr** uniquement.

Cette option permet d'afficher en sortie le script de l'uproc si la variable **U_UPR_SCRIPT_NODISP** a été positionnée à "Y" dans l'environnement courant de la commande (voir manuel d'administration, paragraphe "Affichage du script d'une uproc").

Rappel : par défaut cette variable n'est pas définie et le script de l'uproc est affiché même si l'item SCRIPT est absent.

Exemples

```
uxadd UPR EXP UPR=D_LOAD_FIL LABEL=\"Chargement du fichier\" FPERIOD=d
CLFIL=c:\temp\D_LOAD_FIL.bat APPL="BK" MEMO="O" NBPER=1
```

crée l'uproc D_LOAD_FIL version 000 label "Chargement du fichier" dans l'environnement courant (société, noeud) en espace exploitation, période fonctionnelle : jour, fichier de commande : c:\temp\ D_LOAD_FIL.bat, application BK (domaine T), mémorisée 1 exécution 1 période.

```
uxupd UPR EXP UPR=D_LOAD_FIL RESCON add CONDNO=01 RES=D_FIL_PAY
```

```
uxupd UPR EXP UPR=D_LOAD_FIL RESCON add CONDNO=02 RES=D_BASE LOCK
```

```
uxupd UPR EXP UPR=D_LOAD_FIL FORMULA LFORM=\"=C01 ET =C02\"
```

modifie l'uproc D_LOAD_FIL pour lui définir une formule de lancement composée de deux conditions de ressources en ET, la 1ère sur la présence de la ressource fichier D_FIL_PAY et la 2nde sur la disponibilité totale de la ressource logique D_BASE.

```
uxadd UPR EXP UPR=D_BACKUP LABEL=Sauvegarde FPERIOD=d
CLFIL=c:\temp\D_BACKUP.bat appl=BK
```

crée l'uproc D_BACKUP en espace exploitation label "Sauvegarde" dans l'environnement courant, période fonctionnelle : jour, fichier de commande : c:\temp\D_BACKUP.bat, application BK (domaine T).

```
uxupd UPR EXP UPR=D_BACKUP DEPCON add CONDNO=01 DEP=D_LOAD_FIL
```

```
uxupd UPR EXP UPR=D_BACKUP FORMULA LFORM="=C01"
```

modifie l'uproc D_BACKUP pour lui définir une formule de lancement composée d'une condition d'enchaînement sur l'uproc D_LOAD_FIL (dans l'état Terminé par défaut). Cette uproc libérera dans son script la ressource D_BASE allouée en totalité par l'uproc D_LOAD_FIL.

```
uxupd UPR UPR=D_LOAD_FIL LAUNCHVAR add VARNAME=VARTXT VTYPE=T
VVALUE="XY_VARTXT" VFORMAT="30"
```

crée pour l'uproc D_LOAD_FIL une variable VARTXT de type texte, d'une longueur maximum de 30 caractères, de valeur par défaut « XY_VARTXT ».

```
uxupd UPR UPR=D_LOAD_FIL LAUNCHVAR add VARNAME=VARDATE VTYPE=D
VVALUE="1999/12/31" VFORMAT="AAAA/MM/JJ"
```

crée pour l'uproc D_LOAD_FIL une variable de type date au format année/mois/jour, de valeur par défaut 1999/12/31.

```
uxext UPR EXP UPR=D* output=c:\temp\DOC.ext
```

extrait toutes les uprocs commençant par D_ de l'espace exploitation et archive les données dans le fichier c:\temp\DOC.ext.

```
uxins UPR EXP UPR=* input=c:\temp\DOC.ext
```

insère toutes les uprocs du fichier c:\temp\DOC.ext dans l'environnement courant (société, noeud) en espace exploitation sauf si elles existent déjà.

Les sessions

Les commandes

uxadd SES : crée une session dans la liste des sessions.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	**10c**	
VSES=	3n	
LABEL=	30c	" "
HEADER=	**10c**	
FATHER=	uproc_code(nnn)	
SONOK=	((uproc_code(nnn),{XY}) (uproc_code(nnn),{XY}) (uproc_code(nnn),ug_code)...)	
SONKO=	((uproc_code(nnn),{XY}) (uproc_code(nnn),{XY}) (uproc_code(nnn),ug_code)...)	

Exemples :

```
uxadd SES EXP SES=SES_TEST LABEL=\"Create session\" HEADER=IU_TEST
FATHER=IU_TEST SONOK="((IU_TEST2)(IU_TEST3(001)))"
SONKO="((IU_ALARM(002)))" FATHER=IU_TEST2 SONOK="((IU_TEST3(003),{A
}))" FATHER=IU_TEST3\(003\) SONOK="((D_BACKUP))"
SONKO="((IU_ALARM(004)))"
```

```
uxadd SES EXP SES=D_LOAD_BCK label=\"Load file\" HEADER=D_LOAD_FIL
FATHER=D_LOAD_FIL SONOK="((D_BACKUP))"
```

uxdup SES : duplique une session.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	**10c**	
VSES=	**3n**	

TSES=	**10c**	
TVSES=	**3n**	
TLABEL	60c	

Exemple :

```
uxdup SES EXP SES= TEST VSES=000 TSES=TEST2 TVSES=000 TLABEL=\"Session
de test\"
```

uxshw SES : affiche les caractéristiques d'une session.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	**10c ***	
VSES=	3n *	
LST | LNK		LST
OUTPUT=file_name		

Exemple :

```
uxshw SES EXP SES=SES_TEST LNK
```

uxdlt SES : supprime une session.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	**10c ***	
VSES=	3n *	

Exemple :

```
uxdlt SES EXP SES=SES_TEST
```

uxlst SES : liste les sessions.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	*
VSES=	3n *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst SES OUTPUT=c:\temp\LstSes.txt
```

uxtra SES : transfert les sessions dans un autre espace de la même société sur le même noeud.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD

APP | INT | SIM	3c	S_ESPEXE
SES=	**10c ***	
VSES	3n *	
TESP=	**I | S | X**	

Exemple :

```
uxtra SES app SES=D_* TESP=X
```

uxdst SES : distribue les sessions.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	**10c ***	
VSES	3n *	
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst SES SES=* TMU=S
```

uxext SES : extrait la définition complète d'une session.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	**10c ***	
VSES=	3n *	
REPL		
OUTPUT=file_name		
NOVERIF		

Exemple :

```
uxext SES EXP SES=D_* output=c:\temp\ses.ext
```

uxins SES : insère la définition complète d'une session dans les fichiers correspondants de Dollar Universe pour le même espace et la même version de Dollar Universe que l'extraction.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	**10c ***	
VSES=	3n *	
REPL		
ENABLE | NOENABLE		ENABLE
INPUT=file_name		

Exemple :

```
uxins SES EXP SES=* INPUT=c:\temp\ses.ext
```

Description des items

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée sauf pour **uxadd**).

TSES : Code de la session résultat de la duplication. Format : 10 caractères alphanumériques.

VSES : Version de session. Format : 3 caractères numériques.

Permet la saisie d'une version de session (valeur explicite).

Par défaut, toutes les versions des sessions renseignées sont extraites (inutile avec EXP).

TVSES : version de la session résultat de la duplication. Format : 3 caractères numériques.

LABEL : Libellé descriptif de la session. Format : 30 caractères alphanumériques.

TLABEL : Libellé descriptif de la session résultat de la duplication. Format : 30 caractères alphanumériques.

HEADER : Uproc entête de la session. Format : 10 caractères alphanumériques.

Code de l'uproc en tête de la session.

FATHER : Description de l'origine de la branche décrite. Code de l'uproc mère de la branche.

Remarque : ce code peut être suivi du numéro d'occurrence de l'uproc au sein de la session (3 caractères numériques, ce qui permet de distinguer deux citations de la même uproc dans la session) selon la syntaxe : UPROC_CODE(nnn).

SONOK : Description de l'uproc suivante en chemin normal.

Code de l'uproc fille par rapport au FATHER précédent.

Même remarque que précédemment pour le numéro d'occurrence : cela permet de distinguer plusieurs citations de la même uproc.

Plusieurs uprocs peuvent être décrites sur une même chemin, les codes sont alors présentés entre parenthèses, l'ensemble du chemin est lui même également décrit entre parenthèses, soit par exemple : ((CODE_UPR1)(CODE_UPR2(001))(CODE_UPR3,{XY})(CODE_UPR3,CODE_UG)).

{XY} indique l'expression d'un TIH.

CODE_UG est le code d'une unité de gestion (explicite).

SONKO : Description de l'uproc suivante en chemin anormal.

Code de l'uproc fille par rapport au FATHER précédent, mêmes remarques que précédemment.

REPL : Option de remplacement.

Indique si l'enregistrement extrait (inséré) annule et remplace l'enregistrement déjà existant dans le fichier de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signale par un message pour chaque occurrence).

ENABLE : Option de libération. Valeur par défaut.

Indique si la session sera insérée dans l'état libéré. L'action de libération (06) sera nécessaire avant toute modification de la session si l'objet est inséré avec le qualifier NOENABLE.

LST : Liste des uprocs de la session. Valeur par défaut de la commande **uxshw.**

Indique que le résultat de la commande **uxshw** doit être la liste des uprocs de la session, si une uproc est citée deux fois ou plus dans la session, elle n'apparaîtra qu'une seule fois dans cette liste,

Incompatible avec l'utilisation de LNK.

LNK : Liste des uprocs de la session. Incompatible avec l'utilisation de LST.

Indique que le résultat de la commande **uxshw** doit présenter la session sous la forme d'un tableau décrivant les liens entre les uprocs et le chemin normal ou dégradé.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

OUTPUT : Stockage des résultats. Le nom du fichier est **obligatoire** pour la commande **uxext.**

Si l'item n'est pas renseigné la sortie se fait à l'écran.

INPUT : Fichier d'entrée pour la commande **uxins.** Le nom du fichier est **obligatoire**.

TESP : code de l'espace cible du transfert : X pour Exploitation, S pour Simulation ou I pour intégration.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

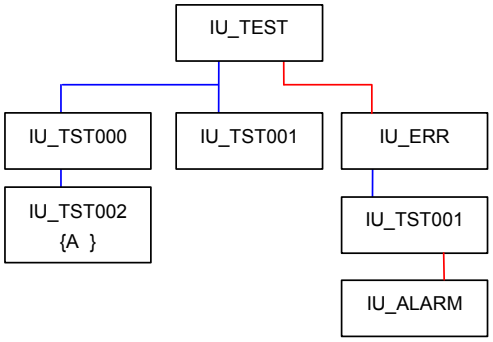
NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

Exemples

```
uxadd SES EXP SES=SES_TEST label=\"Creation session\" HEADER=IU_TEST
FATHER=IU_TEST SONOK=((IU_TST000)(IU_TST001(001))) SONKO=((IU_ERR))
FATHER=IU_TST000 SONOK=((IU_TST002,{A })) FATHER=IU_ERR
SONOK=((IU_TST001(002))) FATHER=IU_TST001(002) SONKO=((IU_ALARM))
```

crée une session "SES_TEST" dans l'environnement courant (société, noeud) selon le schéma suivant :




```
uxdlt SES EXP SES=SES_TEST
```

supprime la session "SES_TEST" dans l'environnement courant (société, noeud).

```
uxadd SES EXP SES=D_LOAD_BCK label=\"Chargement du fichier\"
HEADER=D_LOAD_FIL FATHER=D_LOAD_FIL SONOK=((D_BACKUP))
```

créé la session D_LOAD_BCK composée de l'entête D_LOAD_FIL suivie de l'uproc D_BACKUP en chemin normal sur la même unité de gestion d'exécution.

```
uxshw SES EXP SES=D_LOAD_BCK LNK
```

affiche la session créée précédemment avec les uprocs qui la compose et les liens entre elles.

```
uxext SES EXP SES=D_* output=c:\temp\DOC.ext
```

extrait toutes les sessions commençant par D_ de l'espace exploitation et archive les données dans le fichier c:\temp\DOC.ext.

```
uxins SES EXP SES=* input=c:\temp\DOC.ext
```

insère toutes les sessions du fichier c:\temp\DOC.ext dans l'environnement courant (société, noeud) en espace exploitation sauf si elles existent déjà.

Planification

Les calendriers

Les commandes

uxadd CAL : crée un calendrier dans la liste des calendriers ou une ou plusieurs années sur un calendrier existant.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
MU=	**10c**	
LABEL=	40c	" "
NOMODEL | MODEL		NOMODEL
FRHOLI | NOFRHOLI		FRHOLI
SINCE=	**YYYY**	
BEFORE=	**YYYY**	
WEEK= W C H	7c	WWWWWCC

Exemple :

```
uxadd CAL EXP MU=\" 00\" LABEL=\"General calendar\" SINCE=2001
BEFORE=2011
```

uxupd CAL : modifie les caractéristiques d'un calendrier.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
MU=	**10c ***	
LABEL=	40c	
NOMODEL | MODEL		
DSINCE=	U_FMT_DATE	
DBEFORE=	U_FMT_DATE	
ATTR=W | C | H | ALL		
TO=W | C | H		
DAY=MON | TUE | WED | THU | FRI | SAT | SUN | ALL		ALL

Exemple :

```
uxupd CAL EXP MU=HEADQUART DSINCE=20021226 DBEFORE=20021231 ATTR=ALL
TO=C
```

uxshw CAL : affiche les caractéristiques d'un calendrier.

Item	Format	Défaut

SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
MU=	**10c ***	
NOMODEL | MODEL		
SINCE=	YYYY	
BEFORE=	YYYY	
OUTPUT=file_name		

Exemple :

```
uxshw CAL EXP MU=HEADQUART SINCE=2002 BEFORE=2002
```

uxdlt CAL : supprime un calendrier.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
MU=	**10c ***	
NOMODEL | MODEL		NOMODEL
SINCE=	YYYY	
BEFORE=	YYYY	

Exemple :

```
uxdlt CAL EXP MU=HEADQUART SINCE=1997 BEFORE=2000
```

uxlst CAL : liste les calendriers.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
MU=	10c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst CAL OUTPUT=c:\temp\LstCal.txt
```

uxtra CAL : transfert les calendriers dans un autre espace de la même société sur le même noeud.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
APP | INT | SIM	3c	S_ESPEXE
MU=	**10c ***	
MODEL | NOMODEL		NOMODEL
TESP=	**I | S | X**	

Exemple :

```
uxtra CAL app MU=D_* MODEL TESP=X
```

uxdst CAL : distribue les calendriers modèles.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
EXP | SIM | INT | APP	3c	S_ESPEXE
MU=	**10c ***	
MODEL		
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst UPR UPR=* TMU=S
```

uxext CAL : extrait la définition d'un (modèle de) calendrier (étendue complète).

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UG= | MU=	**10c ***	
REPL		
OUTPUT=file_name		
NOVERIF		

Exemple :

```
uxext CAL EXP MU=HEADQUART output=c:\temp\cal.ext
```

uxins CAL : insère la définition d'un calendrier dans les fichiers correspondants de Dollar Universe (étendue complète) pour le même espace et la même version de Dollar Universe que l'extraction.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
UG= | MU=	**10c ***	
CIBLE=	(MU_CODE,...)	
REPL		
INPUT=file_name		

Exemple :

```
uxins CAL EXP MU=* CIBLE=HEADQUART INPUT=c:\temp\cal.ext
```

Description des items

UG ou **MU** : Code de l'unité de gestion du Calendrier. Format : 10 caractères alphanumériques.

MU permet la saisie d'une unité de gestion (valeur générique acceptée sauf pour **uxadd**).

UG permet la saisie d'un modèle ou d'unité de gestion (commandes **uxext** et **uxins**).

Le calendrier général peut être manipulé par l'intermédiaire de l'unité de gestion " 00" (espace 0 0) ou " " (espace).

MODEL : Type du calendrier. Valeur par défaut : NOMODEL.

Un calendrier de type MODEL devra être inséré ou distribué pour devenir opérationnel. "**UG**" exprime alors le nom du modèle.

LABEL : Libellé descriptif du calendrier. Format : 40 caractères alphanumériques.

SINCE : Première année du calendrier. Format : 4 caractères numériques.

Indique l'année à partir de laquelle le calendrier devra débuter (valeur minimum : année courante moins une).

BEFORE : Dernière année du calendrier. Format : 4 caractères numériques.

Indique la dernière année créée (incluse) du calendrier (valeur minimum : année courante plus deux).

L'écart entre la première et la dernière année du calendrier ne peut dépasser 24 ans.

FRHOLI : Avec jours fériés français. Indique que le calendrier sera créé avec les jours fériés français.

Si le mot clé NOFRHOLI est indiqué, le calendrier sera créé sans les jours fériés français (valeur par défaut).

WEEK : Masque de création.

Indique quel masque de création pour la semaine est utilisé pour créer le calendrier.

Format : pour les sept jours de la semaine (du lundi au dimanche) le type du jour peut être :

- w : ouvré,
- c : chômé,
- h : férié,

soit par exemple : week=(wwwwwcc) : ouvré du lundi au vendredi, samedi et dimanche chômés (valeur par défaut).

DSINCE : jour de début de la modification. Format : indiqué par la variable U_FMT_DATE.

Indique le premier jour à partir de laquelle la modification est appliquée.

DBEFORE : jour de fin de la modification. Format : indiqué par la variable U_FMT_DATE.

Indique le dernier jour (inclus) jusqu'à laquelle la modification est appliquée.

ATTR : sélection sur le type des jours. Indique quel type de jours doit être affecté par la modification.

Format : w (ouvré), c (chômé), h (férié) ou all (valeur par défaut).

TO : Nouveau type. Indique le nouveau type des jours sélectionnés.

Format : w (ouvré), c (chômé), h (férié).

DAY : Sélection en fonction du jour de la semaine.

Indique quel jour dans la semaine (ou quel ensemble de jours) doit être affecté par la modification.

Format :

- MON : les lundis,
- TUE : les mardis,
- WED : les mercredis,
- THU : les jeudis,
- FRI : les vendredis,
- SAT : les samedis,
- SUN : les dimanches,

- ALL : quel que soit le jour dans la semaine (valeur par défaut).

Par exemple : DAY=(SAT,MON) : les samedis et lundis. DAY=WED : les mercredis.

CIBLE : Unités de gestion cibles d'une insertion. Saisie obligatoire. Format : 10 caractères alphanumériques.

Permet d'indiquer, pour les calendriers (obligatoirement de type modèle) sélectionnés, les unités de gestion cibles de l'insertion (démultiplication du modèle en autant de calendriers que d'unités de gestion indiquées dans la cible).

Avec la description de la cible, les modèles visés ne sont pas insérés en tant que modèle de calendrier.

Si deux modèles distincts, par le biais des cibles décrites, donnent lieu à la génération d'un même calendrier (éventuellement dans la même commande), la deuxième insertion est rejetée, tout comme si le calendrier existait au préalable (un message d'avertissement est fourni pour chaque rejet). Il convient donc d'utiliser l'option REPL pour palier ces rejets (c'est alors la dernière insertion qui subsiste).

REPL : Option de remplacement.

Indique si l'enregistrement extrait (inséré) annule et remplace les années déjà existantes dans le fichier (ou le calendrier) de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signale par un message pour chaque occurrence).

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

OUTPUT : Stockage des résultats. Le nom du fichier est **obligatoire** pour la commande **uxext.**

Si l'item n'est pas renseigné la sortie se fait à l'écran.

INPUT : Fichier d'entrée pour la commande uxins. Le nom du fichier est **obligatoire**.

TESP : code de l'espace cible du transfert : X pour Exploitation, S pour Simulation ou I pour intégration.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

Exemples

```
uxadd CAL EXP MU=\" 00\" SINCE=2001 BEFORE=2015 LABEL=\"Calendrier
general\"
```

```
uxadd CAL EXP MU=SIEGE LABEL=\"Calendrier du siege\" SINCE=1995
BEFORE=2015
```

ajoute (ou remplace) les années 2001 à 2015 au calendrier général dans l'environnement courant (société, noeud et espace) et crée le calendrier de l'unité de gestion SIEGE dans l'environnement courant (société, noeud et espace).

```
uxupd CAL EXP MU=SIEGE DSINCE=19991227 DBEFORE=19991231 ATTR=ALL TO=C
```

modifie le calendrier de l'unité de gestion SIEGE de l'environnement courant (société, noeud et espace) pour rendre chômée la période du 27/12/99 au 31/12/99 si la variable U_FMT_DATE vaut YYYYMMDD.

```
uxdlt CAL EXP MU=SIEGE SINCE=1995 BEFORE=1997
```

supprime les années 1995, 1996 et 1997 du calendrier de l'unité de gestion SIEGE de l'environnement courant (société, noeud et espace).

```
uxext CAL EXP MU=MOD output=c:\temp\DOC.ext
```

extrait le calendrier modèle "MOD" de l'espace exploitation et archive les données dans le fichier c:\temp\DOC.ext.

```
uxins CAL EXP MU=MOD CIBLE=SIEGE input=c:\temp\DOC.ext
```

insère le calendrier modèle "MOD" du fichier c:\temp\DOC.ext, comme calendrier de l'unité de gestion SIEGE dans l'environnement courant (société, noeud) en espace exploitation sauf s'il existe déjà.

Les règles de planification

Les commandes

uxadd RUL : crée une règle dans la liste des règles.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RUL=	**8c**	
LABEL=	40c	" "
PERIOD=(n[d | w | m])	4c	(1d)
POS=(±n[w | c])	5c	(+1c)
AUTHO=(ynn,ynn,ynn,ynn,ynn,ynn,ynn)	29c	(ynn,ynn,ynn,ynn,ynn,ynn,ynn)
OFFSET=N | F | P	1c	N

Exemples :

```
uxadd RUL RUL="LWDM" LABEL=\"last working day of the month\"
PERIOD="(1M)" POS="(-1W)"
```

```
uxadd RUL RUL="5DMO+" LABEL=\"the 5th of the month offset +\"
PERIOD="(1M)" POS="(+5C)" OFFSET=F
```

```
uxadd RUL RUL="NO_SUN" LABEL=\"except Sundays\"
AUTHO="(YYY,YYY,YYY,YYY,YYY,YYY,NNN)"
```

uxupd RUL : modifie les caractéristiques d'une règle.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RUL=	**8c ***	
LABEL=	40c	
PERIOD=(n[d | w | m])	4c	
POS=(±n[w | c])	5c	
AUTHO=(ynn,ynn,ynn,ynn,ynn,ynn,ynn)	29c	
OFFSET=N | F | P	1c	

Exemple :

```
uxupd RUL RUL="5DMO+" LABEL=\"the 5th working day of the month offset
+\" POS="(+5W)" OFFSET=N
```

uxshw RUL : affiche les caractéristiques d'une règle.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RUL=	**8c ***	
OUTPUT=file_name		

Exemple :

```
uxshw RUL RUL="5DMO+"
```

uxdlt RUL : supprime une règle.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RUL=	**8c ***	

Exemple :

```
uxdlt RUL RUL=NO_SUN
```

uxlst RUL : liste les règles.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RUL=	8c *	*
INF | FULL		FULL
OUTPUT=file_name		

Exemple :

```
uxlst RUL OUTPUT=c:\temp\LstRul.txt
```

uxext RUL : extrait la définition d'une règle.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RUL=	**8c ***	
REPL		
OUTPUT=file_name		
NOVERIF		

Exemple :

```
uxext RUL RUL=* output=c:\temp\rul.ext
```

uxins RUL : insère la définition d'une règle dans les fichiers correspondants de Dollar Universe pour la même version de Dollar Universe que l'extraction.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RUL=	**8c ***	
REPL		

INPUT=file_name		

Exemple :

```
uxins RUL RUL=* INPUT=c:\temp\rul.ext
```

Description des items

RUL : Code de la règle. Format : 8 caractères : alphanumériques plus les caractères "+" et "-".

Permet la saisie du code de la règle (valeur générique acceptée sauf pour **uxadd**).

LABEL : Libellé descriptif de la règle. Format : 40 caractères alphanumériques.

PERIOD : Périodicité de la règle. Indique quelle est la période de la règle.

Format : un multiple entier de d (jour) w (semaine) ou m (mois).

POS : Position dans la période.

Indique quel numéro de jour est recherché (la numérotation commence à +1 pour le premier jour, à -1 pour le dernier) et le type du jour compté W (ouvré) ou C (calendaire).

Format : entre parenthèses, 2 caractères numériques signés suivis de W ou C. Par exemple : (-1w) : moins un jour ouvré.

AUTHO : Masque d'autorisation de la semaine. Valeurs : y (oui) ou n (non).

Indique pour chaque jour de la semaine (du lundi au dimanche) la validité du calcul de la date en fonction du type de ce jour (ouvré, chômé ou férié).

Format : entre parenthèses, pour chaque jour est indiqué dans un triplet (ouvré chômé férié) si le calcul est valide, chaque jour est séparé du suivant par des virgules. Par exemple la valeur par défaut est : AUTHO=(ynn,ynn,ynn,ynn,ynn,ynn,ynn).

OFFSET : Report. Format : n (aucun), f (suivant) ou p (précédent). Valeur par défaut : n (aucun).

Indique quel report est appliqué si la date calculée n'est pas autorisée dans le masque de la semaine.

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

REPL : Option de remplacement.

Indique si la règle extraite (insérée) annule et remplace la règle existante dans le fichier (ou la liste des règles) de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signalé par un message pour chaque occurrence).

OUTPUT : Stockage des résultats. Le nom du fichier est **obligatoire** pour la commande **uxext.**

Si l'item n'est pas renseigné la sortie se fait à l'écran.

INPUT : Fichier d'entrée pour la commande uxins. Le nom du fichier est **obligatoire**.

NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

Exemples

```
uxadd RUL RUL="DJOMOI" label=\"Dernier jour ouvre du mois\"
period=(1m) pos=(-1w)
```

crée la règle "dernier jour ouvré du mois" de période 1 mois, avec une position sur le dernier jour ouvré sans autorisation ni report particulier, dans l'environnement courant (société et noeud).

```
uxadd RUL RUL="05JM_O" label=\"Le 5eme jour du mois ouvre report +\"
PERIOD=(1m) POS=(+5c) OFFSET=f
```

crée la règle "cinquième jour du mois et report au jour ouvré suivant" de période 1 mois, avec une position sur cinquième jour du mois avec le masque par défaut (jours ouvrés seuls autorisés) et report au suivant, dans l'environnement courant (société et noeud).

```
uxadd RUL RUL=NO_SUN label=\"Sauf les dimanches\" period=(1d)
pos=(+1c) autho=(YYY,YYY,YYY,YYY,YYY,YYY,NNN)
```

crée la règle "tous les jours sauf le dimanche" de période 1 jour, avec une position sur le jour même avec un masque d'autorisation qui refuse le dimanche quel que soit le type de ce jour, dans l'environnement courant (société et noeud).

```
uxlst RUL
```

liste les règles de l'environnement courant (société et noeud).

```
uxdlt RUL RUL=NO_SUN
```

supprime la règle NO_SUN de l'environnement courant (société et noeud).

Les tâches

Les commandes

uxadd TSK : crée une tâche dans la liste des tâches. Cette commande ne crée que l'enveloppe de la tâche, les autres informations doivent être saisie par l'intermédiaire de la commande **uxupd**.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c	" "
VSES=	3n	
UPR=	**10c**	
VUPR=	3n	
MU=	**10c**	
MODEL | NOMODEL		NOMODEL

TECHINF : information générales.

Item	Format	Défaut
TYPE=S | O | P	**1c**	
USER=	**12c**	
PRINTER=	4c	
PRTY=	3n	100
QUEUE=	20c	SYS_BATCH
PDATE=(±n,±n,W | C)		(0,0,C)
HOLD | NOHOLD		NOHOLD
MULT | NOMULT		NOMULT
FORCE | NOFORCE		NOFORCE
CENTRAL | NOCENTRAL		NOCENTRAL
RESTART | NORESTART		NORESTART

Exemple :

```
uxadd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART TECHINF
TYPE=S USER=admin
```

uxupd : modifie les caractéristiques d'une tâche et défini les informations complémentaires lors d'une création.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
MODEL | NOMODEL		NOMODEL

TECHINF : informations générales.

Item	Format	Défaut
TYPE=S | O | P	1c	
USER=	12c	
PRINTER=	4c	
PRTY=	3n	
QUEUE=	20c	
PDATE=(±n,±n,W | C)		
HOLD | NOHOLD		
MULT | NOMULT		
FORCE | NOFORCE		
CENTRAL | NOCENTRAL		
RESTART | NORESTART		

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART TECHINF
USER=univ50e PDATE="(-1,0,W)"
```

IMPL add : ajoute une règle de planification à la tâche.

Item	Format
RULE=	((RULE_CODE,U_FMT_DATE)(…,…))
SDATE=	**U_FMT_DATE**

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART IMPL add
rule="(("LWDM",20020701)("5DMO+",20020701))"
```

IMPL dlt : supprime une règle de planification de la tâche.

Item	Format
RULE=	((RULE_CODE,U_FMT_DATE)(…,…))

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART IMPL dlt
rule="("5DMO+",20020701)"
```

EXPLI add : ajoute une date explicite à la tâche.

Item	Format
EDATE=	((U_FMT_DATE,hhmm,hhh,mm,U_FMT_DATE) (…,…,…,…,…))

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART EXPLI add
EDATE="((20021230,2100,005,00,20021231)(20021130,2100,005,00,20021130)
)"
```

EXPLI dlt : supprime une date explicite de la tâche.

Item	Format
EDATE=	((U_FMT_DATE,hhmm,hhh,mm,U_FMT_DATE) (…,…,…,…,…))

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART EXPLI dlt
EDATE="(20021130,2100,005,00,20021130)"
```

EXCLU add : ajoute une date d'interdiction à la tâche.

Item	Format
XDATE=	((U_FMT_DATE)(…))

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART EXCLU add
XDATE="((20021231)(20030101))"
```

EXCLU dlt : supprime une date d'interdiction de la tâche.

Item	Format
XDATE=	((U_FMT_DATE)(…))

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART EXCLU dlt
XDATE="((20021231))"
```

TIME : horaire de planification journalier.

Item	Format
DALW=	**(hhmm,hhh,mm)**
MON=	(hhmm,hhh,mm)
TUES=	(hhmm,hhh,mm)
WED=	(hhmm,hhh,mm)
THU=	(hhmm,hhh,mm)
FRI=	(hhmm,hhh,mm)
SAT=	(hhmm,hhh,mm)
SUN=	(hhmm,hhh,mm)

Exemple :

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART TIME
DALW="(2100,005,00)" WED="(1900,005,00)"
```

TIME : horaire de planification multi journalier.

Item	Format

MULW=	((hhmm,hhmm,mmm),hhh,mm)

Exemple :

```
uxupd TSK EXP UPR=D_LOAD_FIL MU=HEADQUART TIME
MULW="((2000,2359,030),000,15)"
```

TIME : horaire de planification d'une tâche provoquée.

Item	Format
PRLW=	(hhmm,hhh,mm, hhmm,hhmm)

Exemple :

```
uxupd TSK EXP UPR=D_LOAD_FIL MU=HEADQUART TECHINF TYPE=P

uxupd TSK EXP UPR=D_LOAD_FIL MU=HEADQUART TIME
PRLW="(0800,012,00,1300,1400)"
```

LAUNCHVAR add : ajoute la valorisation d'une variable.

Item	Format
VARNAME=	**20c**
VVALUE=	dépend de la définition de la variable

Exemple :

```
uxupd TSK UPR=D_LOAD_FIL MU=HEADQUART LAUNCHVAR add VARNAME=VARTXT
VVALUE=XY_VARTXT_VALTSK
```

LAUNCHVAR dlt : supprime la valorisation d'une variable.

Item	Format
VARNAME=	**20c**

Exemple :

```
uxupd TSK UPR=D_LOAD_FIL MU=HEADQUART LAUNCHVAR dlt VARNAME=VARTXT
```

uxdup TSK : duplique une tâche.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c	
VSES=	3n	
UPR=	**10c**	
VUPR=	**3n**	
MU=	**10c**	
MODEL | NOMODEL		NOMODEL
TSES=	10c	
TVSES=	3n	
TUPR=	**10c**	
TVUPR=	**3n**	
TMU=	**10c**	
TMODEL | TNOMODEL		TNOMODEL

Exemple :

```
uxdup TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART
SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART MODEL
```

uxshw TSK : affiche les caractéristiques d'une tâche.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
MODEL | NOMODEL		NOMODEL
OUTPUT=file_name		

Exemple :

```
uxshw TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART|more
```

uxdlt TSK : supprime une tâche.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
MODEL | NOMODEL		NOMODEL

Exemple :

```
uxdlt TSK EXP UPR=D_LOAD_FIL MU=HEADQUART
```

uxlst TSK : liste les tâches.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	*
VSES=	3n *	*
UPR=	10c *	*
VUPR=	3n *	*
MU=	10c *	*
MODEL | NOMODEL		NOMODEL
INF | FULL		INF
OUTPUT=file_name		

Exemple :

```
uxlst TSK FULL OUTPUT=c:\temp\LstTsk.txt
```

uxtra TSK : transfert les tâches dans un autre espace de la même société sur le même noeud.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
APP | INT | SIM	3c	S_ESPEXE
SES=	10c *	
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
MODEL | NOMODEL		NOMODEL
TESP=	**I | S | X**	

Exemple :

```
uxtra TSK app UPR=D_* VUPR=* SES=D_* VSES=* MU=* TESP=X
```

uxdst TSK : distribue les tâches modèles.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NŒUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES	3n *	
UPR=	**10c ***	
VUPR	3n *	
MU=	**10c ***	
MODEL		
TMU=	**10c ***	
TMUT=	**1c ***	

Exemple :

```
uxdst TSK EXP UPR=* VUPR=* MU=* MODEL TMU=S
```

uxext TSK : extrait la définition complète d'une tâche (modèle ou non).

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	*
VSES=	3n *	
UPR=	**10c ***	*
VUPR=	3n *	
MU=	**10c ***	*
MODEL		Non modèle

REPL **OUTPUT=file_name** NOVERIF		

Exemple :

```
uxext TSK EXP MU=H* UPR=D_* OUTPUT=c:\temp\tsk.ext
```

uxins TSK : insère la définition d'une tâche (modèle ou non) dans les fichiers correspondants de Dollar Universe pour le même espace et la même version de Dollar Universe que l'extraction et dans l'état Activé/Désactivé qu'elle possédait au moment de son extraction.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
MODEL		Non modèle
CIBLE=	**(MU_CODE,MU_CODE,…) ***	
REPL **INPUT=file_name**		

Exemples :

```
uxins TSK EXP MU=* UPR=* CIBLE=HEADQUART INPUT=c:\temp\tsk.ext
uxins TSK EXP MU=* UPR=* MODEL CIBLE=model INPUT=c:\temp\tsk.ext
```

Description des familles

Une seule des familles décrites ci-dessous peut être utilisée dans une commande uxupd.

TECHINF : Informations techniques

Permet la mise à jour des informations techniques de la tâche (compte de soumission, queue batch...).

TIME : Horaires de lancement

Permet la définition des heures de lancement journaliers ou multi journaliers pour des tâches planifiées ou optionnelles ainsi que les heures de lancement ou d'exclusion d'une tâche provoquée.

IMPL : Planification implicite

Permet la définition de règles de planification et de leurs dates d'application ainsi que la date de première planification de la tâche.

EXPLI : Dates explicite

Permet la définition de dates et heures explicites de planification de la tâche ainsi que la date de traitement associée.

EXCLU : Dates d'interdiction

Permet la définition de dates d'interdiction visant le calcul de la planification à partir des règles définies pour la tâche.

LAUNCHVAR : Variables d'une tâche

Permet la création et la suppression des variables dans la tâche et leur valorisation. Ces variables doivent être choisies dans la liste des variables de l'uproc identifiant la tâche.

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée sauf pour **uxadd**) **obligatoire.**

TUPR: Code de l'uproc de la tâche résultat de la duplication. Format : 10 caractères alphanumériques.

VUPR : Version d'uproc. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une uproc.

Par défaut, toutes les versions des uprocs renseignées sont considérées (inutile avec SIM et EXP).

TVUPR : version de l'uproc de la tâche résultat de la duplication. Format : 3 caractères numériques.

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée sauf pour **uxadd**).

TSES : Code de la session de la tâche résultat de la duplication. Format : 10 caractères alphanumériques.

VSES : Version de session. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une session.

Par défaut, toutes les versions des sessions renseignées sont considérées (inutile avec EXP).

TVSES : version de la session de la tâche résultat de la duplication. Format : 3 caractères numériques.

UG ou **MU** : Code du modèle ou de l'unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**), saisie **obligatoire**.

TMU : code de l'unité de gestion de la tâche résultat de la duplication. Format : 10 caractères alphanumériques.

TYPE : Type de la tâche. Format : S (planifiée), O (optionnelle) ou P (provoquée).

MODEL : Nature de la tâche. Indique que la tâche est modèle. Une tâche non modèle ne porte pas de mot clé.

Dans le cas de la commande **uxins**, MODEL indique que la tâche doit être insérée comme modèle. Si rien n'est spécifié, la tâche sera insérée en non modèle, quelle que soit sa nature d'origine.

TNOMODEL : Nature de la tâche résultat de la duplication. Indique que la tâche dupliquée doit être non modèle.

Pour dupliquer une tâche en modèle, le mot clé TMODEL doit être utilisé.

PDATE : Algorithme de calcul de la date de traitement.

Permet de définir automatiquement une date de traitement différente de la date de traitement qui aurait été calculée par l'automate.

Format : (J,U,T) où :

J : est le nombre de jours, signé, pour une uproc avec période fonctionnelle, 0 sinon,

U : est le nombre d'unités de la période fonctionnelle, si la période est jour, ce nombre vaut 0,

T est le type de décalage en jours ouvrés : W ou calendaires : C.

Valeur par défaut : la date de traitement est calculée par rapport au jour du premier pilotage du lancement.

Par exemple : PDATE=(-1,0,C) : moins un jour calendaire du même mois.

PRINTER : Imprimante. Non utilisé sous UNIX, Windows et VMS

PRTY : Priorité. Format : 3 caractères numériques de 001 à 255. Valeur par défaut : 100.

Priorité d'entrée de la tâche dans la queue batch.

QUEUE : Queue batch. Format : 20 caractères alphanumériques ("$" accepté sous VMS).

Queue batch de soumission de la tâche. Valeur par défaut : SYS_BATCH.

USER : Compte de soumission. Format : 12 caractères alphanumériques.

Compte dans l'environnement duquel l'exécution sera soumise.

HOLD : Désactiver une tâche. Valeur par défaut : NOHOLD.

Indique que la tâche sera désactivée : les lancements générés à partir de la tâche seront donc désactivés.

FORCE : Lancement forcé en fin de période. Valeur par défaut : NOFORCE.

Indique que l'exécution du lancement sera forcée en fin de plage horaire de la tâche même si toutes les conditions de l'uproc ne sont pas vérifiées.

CENTRAL : Contrôle centralisé. Autre valeur : NOCENTRAL.

Valeur par défaut : indiquée par la variable S_CODCENTRAL.

Indique que les différents états d'exécution de la tâche seront remontés au noeud de contrôle centralisé.

RESTART : Relance automatique. Valeur par défaut : NORESTART.

Indique, dans le cas de l'utilisation d'un gestionnaire de queues batch uniquement, que le lancement sera automatiquement relancé lors du redémarrage de la machine s'il était en cours d'exécution lors de l'arrêt de celle-ci.

MULT : Lancements multiples.

Indique que les horaires de lancements sont multi journaliers. La famille TIME doit alors être complété par le mot clé MULW.

Valeur par défaut : NOMULT. La famille TIME est alors complété par le mot clé DALW pour une tâche planifiée ou PRLW pour une tâche provoquée.

CIBLE : Unités de gestion cibles de l'insertion. Limité à **uxins**.

Format :

- * : valeur générique acceptée,
- 10 caractères alphanumériques, si la cible est unique,
- (CODE_UG,CODE_UG,…) si la cible est multiple, CODE_UG étant un code d'unité de gestion (10 caractères alphanumériques).

Si deux tâches distinctes, par le biais des cibles décrites, donnent lieu à l'insertion d'une même tâche (éventuellement dans la même commande), la deuxième insertion est rejetée, tout comme si la tâche existait déjà (un message d'avertissement est fourni pour chaque rejet). Si l'option REPL est utilisée, la dernière insertion écrase la première.

Si CIBLE=*, les tâches non modèles contenues dans le fichier d'import sont insérées avec leur UG de départ. L'UG de la tâche contenue dans le fichier d'import est contrôlée dynamiquement et doit exister sur la machine cible. Si elle n'existe pas, la tâche n'est pas insérée. Cette anomalie ne bloque pas la suite du processus d'insertion pour les autres tâches contenue dans le fichier d'import.

Cette fonctionnalité permet de réaliser des opérations de backup / restore pour l'ensemble de la planification active, par les 4 commandes ci-dessous (il est prudent de réaliser une extraction pour les tâches modèles et une autre pour les tâches non modèles) :

```
uxext TSK SES=* UPR=* MU=* OUTPUT=c:\temp\tsk_nomod.ext
```

```
uxins TSK SES=* UPR=* MU=* CIBLE=* INPUT=c:\temp\tsk_nomod.ext
uxext TSK SES=* UPR=* MU=* MODEL OUTPUT=c:\temp\tsk_mod.ext
uxins TSK SES=* UPR=* MU=* MODEL CIBLE=* INPUT=c:\temp\tsk_mod.ext
```

REPL : Option de remplacement.

Indique si l'enregistrement extrait (inséré) annule et remplace l'enregistrement déjà existant dans le fichier (ou la tâche) de destination; par défaut, l'extraction (l'insertion) est refusée pour chaque enregistrement déjà existant (signale par un message pour chaque occurrence).

INF : Option d'affichage. Limite l'affichage au nom des objets. Autre valeur : FULL (par défaut).

Affiche toutes les informations disponibles sur la liste des objets.

DALW : Horaires de lancement journaliers. Doit être associé au mot clé NOMULT.

Indique le début de la plage horaire de lancement et sa durée. Une plage standard peut être définie quel que soit le jour de la semaine (mot clé : DALW), des exceptions peuvent être faites pour chaque jour (mots clés : MON, TUE, WED, THU, FRI, SAT, SUN)

Valeur par défaut indiquée par la variable S_PLAGEHORAIRE.

Format : DALW=(hhmm,hhh,mm) :

- HHMM : heure de début de la plage horaire en heure et minutes,
- HHH : durée de la plage horaire en heures.
- MM : durée de la plage horaire en minutes.

Le nombre de caractères pour la saisie des heures et des minutes doit être respecté exactement.

MULW : Horaires de lancement multi journaliers. Doit être associé au mot clé MULT.

Permet de définir des plages multiples de lancement dans la journée.

Format : MULW=((hhmm,hhmm,mmm),hhh,mm) :

- HHMM,HHMM : heure de début et de fin du calcul en heure et minutes,
- MMM : périodicité du calcul,
- HHH : durée de la plage horaire en heures.
- MM : durée de la plage horaire en minutes.

le module (HHMM,HHMM,MMM) peut être répété autant de fois que nécessaire.

Le nombre de caractères pour la saisie des heures et des minutes doit être respecté exactement.

PRLW : Horaires de lancement d'une tâche provoquée. Doit être associé au mot clé NOMULT.

Permet de définir une plage horaire de lancement et d'exclusion.

Par défaut aucune plage horaire n'est définie, la tâche sera soumise dès sa provocation.

Format : PRLW=(hhmm,hhh,mm,hhmm,hhmm) :

- HHMM: heure de début de soumission immédiate,
- HHH : durée de la plage horaire en heures,
- MM : durée de la plage horaire en minutes,
- HHMM,HHMM : heure de début et de fin de soumission différée.

Le nombre de caractères pour la saisie des heures et des minutes doit être respecté exactement.

RULE : Règle de planification de la tâche.

Permet d'associer une ou plusieurs nouvelles règles de planification à la tâche ou d'en supprimer.

Format : ((COD_REGLE,A_DATE)(COD_REGLE, A_DATE)…) où :

- COD_REGLE est le code de la règle manipulée,
- A_DATE est la date d'application de la règle à la tâche au format indiquée par la variable U_FMT_DATE.

le module (COD_REGLE, A_DATE) peut être répété au maximum sept fois dans la même commande.

le nombre de caractères A_DATE doit être respecté exactement.

SDATE : Date de première planification. Format : indiqué par la valeur de la variable U_FMT_DATE.

Indique à partir de quelle date la tâche devra être planifiée.

EDATE : Dates explicites. Permet d'ajouter ou de supprimer une date explicite à la tâche. Format :

((E_DATE,HHMM,HHH,MM,P_DATE)…) où :

- E_DATE est la date explicite de lancement, au format indiqué par la valeur de la variable U_FMT_DATE,
- HHMM est l'heure de début de la plage horaire,
- HHH est la durée de la plage horaire en heures,
- MM est la durée de la plage horaire en minutes,
- P_DATE est la date de traitement associée à l'exécution, au format indiqué par la valeur de la variable U_FMT_DATE.

le module (E_DATE,HHMM,HHH,MM,P_DATE) peut être répété autant de fois, dans la même commande, que des dates explicites doivent être ajoutées ou supprimées.

Le nombre de caractères doit être respecté exactement.

XDATE : Dates d'interdiction. Permet d'ajouter ou de supprimer une date d'interdiction à la tâche.

Format : ((X_DATE)…) où :

X_DATE : est la date d'interdiction de lancement, au format indiqué par la valeur de la variable U_FMT_DATE,

le module (X_DATE) peut être répété autant de fois, dans la même commande, que des dates d'interdiction doivent être ajoutées ou supprimées.

Le nombre de caractères doit être respecté exactement.

VARNAME : Nom de la variable. Format : 20 caractères alphanumériques maximum. Saisie obligatoire.

Ce nom doit être choisi dans la liste des variables de l'uproc identifiant la tâche.

VVALUE : Valeur de la variable dans la tâche.

Si la valeur par défaut de la variable de l'uproc doit être modifiée, la saisie de VVALUE est requise,

Le format doit être compatible avec le format déclaré lors de la création de la variable dans l'uproc, la valeur doit être comprise dans la plage définie pour une variable de type quantité.

OUTPUT : Stockage des résultats. Le nom du fichier est **obligatoire** pour la commande **uxext.**

Si l'item n'est pas renseigné la sortie se fait à l'écran.

INPUT : Fichier d'entrée pour la commande uxins. Le nom du fichier est **obligatoire**.

TESP : code de l'espace cible du transfert : X pour Exploitation, S pour Simulation ou I pour intégration.

TMU : code de l'unité de gestion cible de la distribution. Format : 10 caractères alphanumériques.

(valeur générique acceptée).

TMUT : code du type de l'unité de gestion cible de la distribution. Format : 1 caractère alphanumérique.

(valeur générique acceptée).

NOVERIF : peut être utilisé avec la commande **uxext** uniquement.

Cette option permet de ne pas vérifier le fichier l'extraction avant d'y écrire de nouveaux enregistrements et d'améliorer ainsi les performances de cette commande. Cet item ne doit être utilisé que pour un nouveau fichier.

Exemples

```
uxadd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=SIEGE TECHINF type=S
USER=admin
```

crée une tâche à partir de la session D_LOAD_BCK, uproc entête D_LOAD_FIL sur l'unité de gestion SIEGE non modèle dans l'espace d'exploitation. La tâche est planifiée, le compte de soumission est admin. Cette tâche ne peut pas créer de lancement car, dans l'état, aucune date n'a été définie.

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=SIEGE TECHINF PDATE=(-
1,0,W)
```

met à jour la tâche créée ci-dessus en indiquant que la calcul de la date de traitement est "moins un jour ouvré" par rapport à la date de premier pilotage du lancement. Les autres paramètres ont conservé leur valeur.

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=SIEGE TIME
DALW=(2100,005,00) WED=(1900,005,00)
```

met à jour la tâche créée ci-dessus en indiquant des heures de lancement : tous les jours à partir de 21h pendant 5 heures sauf le mercredi à partir de 19h pendant 5 heures.

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=SIEGE IMPL ADD
RULE=(("DJOMOI",19990501)("05JM_O",19990501))
```

met à jour la tâche créée ci-dessus en indiquant une planification selon les deux règles : "DJOMOI" (dernier jour ouvré du mois) et "05JM_O" (5ème jour du mois report au jour ouvré suivant) avec des dates d'application prévues au 1er mai 1999, dans le cas où la variable U_FMT_DATE vaut YYYYMMDD.

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=SIEGE EXPLI ADD
EDATE=((19991230,2100,005,00,19991231))
```

met à jour la tâche créée ci-dessus en créant une date explicite pour le 1er janvier 1997 à 21 heures pendant 5 heures avec la date de traitement du 2 janvier 97, dans le cas où la variable U_FMT_DATE vaut YYYYMMDD.

```
uxupd TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=SIEGE EXCLU ADD
XDATE=((19991231))
```

met à jour la tâche créée ci-dessus en créant une date d'interdiction pour le 31 décembre 1996, dans le cas où la variable U_FMT_DATE vaut YYYYMMDD.

```
uxadd TSK EXP UPR=D_LOAD_FIL MU=SIEGE TECHINF type=P USER=admin
```

crée une tâche provoquée de l'uproc D_LOAD_FIL sur l'unité de gestion SIEGE non modèle dans l'espace d'exploitation. Le compte de soumission est admin.

```
uxlst TSK EXP UPR=D_* FULL
```

liste les tâches qui viennent d'être créées ainsi que leurs caractéristiques.

```
uxupd TSK UPR=D_LOAD_FIL MU=SIEGE LAUNCHVAR ADD VARNAME=VARTXT
VVALUE=XY_VARTXT_VALTSK
```

modifie la valeur de la variable VARTXT définie pour D_LOAD_FIL en donnant une valeur par défaut de «XY_VARTXT_VALTSK» pour cette variable dans la tâche.

uxupd TSK UPR=D_LOAD_FIL MU=SIEGE LAUNCHVAR ADD VARNAME=VARDATE VVALUE=2000/01/01

modifie la valeur de la variable VARDATE définie pour D_LOAD_FIL en donnant une valeur par défaut de « 2000/01/01 » pour cette variable dans la tâche.

uxext TSK EXP MU=MOD UPR=D_* MODEL output=c:\temp\DOC.ext

extrait toutes les tâches modèles "MOD" de l'espace exploitation dont l'uproc commence par D_ et archive les données dans le fichier c:\temp\DOC.ext.

uxins TSK EXP MU="*" UPR=* CIBLE=SIEGE input=c:\temp\DOC.ext

insère toutes les tâches (non modèles par défaut) du fichier c:\temp\DOC.ext, comme tâches de l'unité de gestion SIEGE dans l'environnement courant (société, noeud) en espace exploitation sauf si elles existent déjà.

uxins TSK EXP MU=* UPR=* CIBLE=* MODEL input=c:\temp\DOC.ext

insère toutes les tâches du fichier c:\temp\DOC.ext, comme tâches modèles dans l'environnement courant (société, noeud) en espace exploitation sauf si elles existent déjà.

Les commandes d'exploitation

Les objets de paramétrage

Les ressources

Les commandes

La ressource doit avoir fait l'objet d'un pilotage avec des quotas avant de pouvoir exécuter l'une de ces commandes.

uxhld RES : alloue tout ou partie des quotas d'une ressource.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	
LOCK | QT1= & QT2=		
QT1=	4n	0
QT2=	4n	0
MU=	10c * " "	*
PDATE=	YYYYMMDD * " "	*
ESP=	X | S | I | A | * " "	*
TSOC=	6c * " "	*

Exemple :

```
uxhld RES RES=D_BASE QT1=4
```

uxrls RES : libère tout ou partie des quotas d'une ressource.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	
CLEAR | QT1= & QT2=		
QT1=	4n	0
QT2=	4n	0
MU=	10c * " "	*

PDATE=	YYYYMMDD * " "	*
ESP=	X | S | I | A | * " "	*
TSOC=	6c * " "	*

Remarque : la commande **uxrls res** étant susceptible de traiter l'ensemble des espaces, les tentatives de connexion à des serveurs d'IO arrêtés peut faire perdre du temps. Le positionnement de la variable d'environnement U_RES_AVAIL_AREAS, avec le code des espaces effectivement actifs permet d'optimiser les performances. Voir manuel d'administration.

Exemple :

```
uxrls RES RES=D_BASE CLEAR
```

uxshw RES : affiche la définition d'une ressource et l'allocation de ses quotas.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	
TSOC=	6c *	
MU=	10c *	
PDATE=	YYYYMMDD * " "	
ESP=	X | S | I | A | * " "	
INF | FULL | RSV		INF
OUTPUT=file name		

Exemple :

```
uxshw RES RES=D_*
```

uxrsv RES : réserve les quotas d'une ressource afin de définir une priorité d'ordonnancement.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c**	
VUPR=	3n	
SES=	10c	
VSES=	3n	
MU=	**10c**	
QT1=	4n	
QT2=	4n	
ESP=	X | S | I | A | * " "	
TSOC=	6c * " "	
PDATE=	YYYYMMDD * " "	

Exemple :

```
uxrsv RES RES=D_FIL_PAY ESP=X UPR=TEST_ECA MU=$S_NOEUD QT1=10 QT2=0
PDATE=$S_DATRAIT
```

Attention : la variable S_DATRAIT est définie en interne dans le script de l'uproc.

uxcnr RES : supprime la réservation des quotas d'une ressource faite par la commande **uxrsv**.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	
EXP | SIM | INT | APP	3c	S_ESPEXE
UPR=	**10c ***	
VUPR=	3n	
SES=	10c	
VSES=	3n	
MU=	**10c**	
TSOC=	6c * " "	
ESP=	X | S | I | A | * " "	
PDATE=	YYYYMMDD * " "	

Exemple :

```
uxcnr RES RES=D_FIL_PAY ESP=X UPR=TEST_ECA MU=$S_NOEUD
PDATE=$S_DATRAIT
```

Attention : la variable S_DATRAIT est définie en interne dans le script de l'uproc.

uxpre RES : supprime des événements ressource dont les quotas ont été libérés (la ressource peut être variabilisé avec !UG!, !PDATE!, !SOC! ou !ESP!).

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
RES=	**20c ***	
EXP | SIM | INT | APP	3c	S_ESPEXE
MU=	10c *	
PDATE=	YYYYMMDD * " "	
TSOC=	6c * " "	
ESP=	X | S | I | A | * " "	

Description des items

RES : Référence ressource. Format : 20 caractères alphanumériques.

Requiert la saisie de la référence de la ressource (valeur générique acceptée pour **uxshw** avec l'option FULL).

TSOC : Société. Format : 6 caractères alphanumériques.

La saisie du caractère "*" permet la sélection de toutes les ressources faisant référence à une société, la saisie du caractère " " permet la sélection de toutes les ressources ne faisant pas référence à une société.

Permet la sélection des ressources faisant référence à une société dans le nom du fichier ou sa localisation.

ESP : Espace. Format : 1 caractère : "X", "S", "I", "A", "*" ou " ".

La saisie du caractère "*" permet la sélection de toutes les ressources faisant référence à un espace, la saisie du caractère " " permet la sélection de toutes les ressources ne faisant pas référence à un espace

Permet la sélection des ressources faisant référence à un espace dans le nom du fichier ou sa localisation.

Pour les commandes **uxrsv** et **uxcnr**, permet de spécifier dans quel espace se trouve la tâche ayant fait la réservation, l'espace décrit par la variable S_ESPEXE est pris par défaut.

UPR : Code uproc. Commandes **uxrsv** et **uxcnr** uniquement. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc, identifiant de la tâche qui réserve ou libère la ressource (valeur générique acceptée sauf pour **uxrsv**) **obligatoire**.

VUPR : Version d'uproc. Commandes **uxrsv** et **uxcnr** uniquement. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une uproc. Par défaut, toutes les versions des uprocs renseignées sont considérées (inutile avec SIM et EXP).

SES : Code session. Commandes **uxrsv** et **uxcnr** uniquement. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session , identifiant de la tâche qui réserve ou libère la ressource (valeur générique acceptée sauf pour **uxrsv**).

VSES : Version de session. Commandes **uxrsv** et **uxcnr** uniquement. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une session. Par défaut, toutes les versions des sessions renseignées sont considérées (inutile avec EXP).

MU : Unité de gestion. Format : 10 caractères alphanumériques.

Permet la sélection des ressources faisant référence à une unité de gestion dans le nom du fichier ou sa localisation.

La saisie du caractère "*" permet la sélection de toutes les ressources faisant référence à une UG, la saisie du caractère " " permet la sélection de toutes les ressources ne faisant pas référence à une UG.

Pour les commandes **uxrsv** et **uxcnr**, permet d'identifier la tâche qui réserve ou libère la ressource.

PDATE : Date de traitement.

Permet la sélection des ressources faisant référence à une date de traitement dans le nom du fichier ou sa localisation.

Format : AAAAMMJJ. La saisie du caractère "*" permet la sélection de toutes les ressources faisant référence à une date de traitement, la saisie du caractère " " permet la sélection de toutes les ressources ne faisant pas référence à une date de traitement

Pour les commandes **uxrsv** et **uxcnr**, permet d'identifier la tâche qui réserve ou libère la ressource.

QT1 ou **QT2** : Quotas d'allocation de niveau 1 et 2 respectivement. Format : de 0 à 9999.

Dans la commande **uxhld**, ces quotas permettent d'allouer les valeurs indiquées, lors de l'exécution de la commande, dans la mesure où ils sont disponibles. L'utilisation de QT1 ou QT2 est incompatible avec l'utilisation de LOCK.

Dans la commande **uxrsv**, ces quotas permettent de réserver, lors de l'exécution de la commande, pour la tâche spécifiée, les valeurs indiquées dans la mesure où ces quotas sont disponibles.

Dans la commande **uxrls**, ces quotas permettent de libérer les valeurs de quotas indiquées lors de l'exécution de la commande. L'utilisation de QT1 ou QT2 est incompatible avec l'utilisation de CLEAR.

CLEAR : Libération. Permet de libérer la ressource sélectionnée,.

Utilisé dans **uxrls**, la ressource est totalement libérée sans distinction de l'origine de l'allocation par l'exécution de la commande. L'usage de CLEAR est incompatible avec QT1 ou QT2.

FULL : Affichage de l'allocation. INF, FULL et RSV sont incompatibles entre eux.

Permet l'affichage de la totalité des informations sur l'allocation d'une ressource : uproc origine de l'allocation, quotas alloués par une uproc, quotas alloués par les commandes, quotas disponibles.

Autre valeur RSV : permet l'affichage des réservations de la ressource.

Autre valeur INF : se reporter au chapitre "Les commandes de paramétrage" paragraphe "Les ressources" page 38.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items.

Exemples

uxhld RES RES=D_BASE QT1=4

alloue un quota de niveau 1 pour une valeur de 4 sur la ressource "D_BASE" dans l'environnement courant (société, noeud).

uxshw RES RES=D_BASE FULL

affiche les allocations de la ressource D_BASE dans l'environnement courant (société, noeud). Soit par exemple :

```
Dde Ress: D_BASE   * *
Utilisation des quotas de la ressource
Ressource: D_BASE      Ress:   Exec:
quotas maxi     Alloc. Auto.    Alloc. Manu.     Residuel
0010    0010     0000   0000     0004   0000     0006   0010
```

quotas maxi : affiche les valeurs maximum des quotas 1 et 2,

Alloc. Auto. : affiche les valeurs des quotas 1 et 2 alloués par des uprocs en cours d'exécution,

Alloc. Manu. : affiche les valeurs des quotas 1 et 2 alloués par des uprocs terminées ou par des commandes **uxhld RES**,

Résiduel : affiche les valeurs des quotas 1 et 2 restantes pour des allocations futures.

Si la même commande est exécutée pendant l'exécution de l'uproc D_LOAD_FIL (qui a une condition de ressource sur la totalité de D_BASE) sur l'unité de gestion et l'espace d'exécution, on obtient alors le résultat suivant :

```
Dde Ress: D_BASE   * *
Utilisation des quotas de la ressource
Ressource: D_BASE      Ress:   Exec:
quotas maxi     Alloc. Auto.    Alloc. Manu.     Residuel
0010    0010     0000   0000     0004   0000     0006   0010
```

```
Utilisation des quotas de la ressource
Ressource: D_BASE  X SIEGE  Ress:   Exec: EXCL
session     Uproc      Es UG        Date Trt    Quo1 Quo2 X Numproc
D_LOAD_BCK D_LOAD_FIL X SIEGE                  0010 0010 O 0000259
quotas maxi     Alloc. Auto.    Alloc. Manu.     Residuel
0010    0010     0010   0010     0000   0000     0000   0000
```

La ressource D_BASE est décomposée en fait en 2 ressources (symbolisé dans l'exemple par les deux cadres) en fonction de son environnement d'exécution. La première allocation résulte de l'exécution de la commande **uxhld**, la seconde allocation résulte de l'exécution de l'uproc D_LOAD_FIL.

Ressource : indique la référence de la ressource suivi éventuellement de l'espace, la date de traitement, l'unité de gestion ou la société si les mots clés correspondants sont intégrés au nom de la ressource. C'est le cas ici pour D_BASE dont le nom est défini par : !UG!!ESP!.

Exec : EXCL indique que la ressource a été attribuée en exclusive à l'exécution de l'uproc.

session Uproc Es UG Date Trt Quo1 Quo2 X Numproc : indiquent le contexte dans lequel la ressource a été attribuée. **Es** pour espace, **Date Trt** pour Date de traitement, **Quo** pour quota.

uxrls RES RES=D_BASE ESP=* MU=* CLEAR

libère tous les quotas de la ressource D_BASE allouée au titre de toutes les unités de gestion et espaces.

```
uxrsv RES RES=PRIO_ORDO ESP=X UPR=IU_DT1 MU=S01 QT1=90 QT2=0
PDATE=${S_DATRAIT}
```

réserve, pour la tâche définie par l'uproc IU_DT1, l'unité de gestion S01 dans l'espace exploitation, pour la date de traitement courante, la ressource de référence PRIO_ORDO avec un quota QT1 de 90.

Si la ressource dispose d'un quota QT1 maximum de 100, 10 restent libres à l'issue de l'exécution de la commande. Une autre tâche, n'ayant pas effectué de réservation sur cette ressource, pourra s'exécuter si sa condition sur cette ressource demande un quota inférieur à 10.

```
uxrsv RES RES=PRIO_ORDO ESP=X UPR=IU_DT2 MU=S01 QT1=10 QT2=0
```

Ces deux dernières commandes définissent une priorité de pilotage par Dollar Universe, c'est la tâche qui a réservé le plus de quotas qui s'exécutera en premier, dans cet exemple : celle définie sur l'uproc IU_DT1.

```
uxcnr RES RES=PRIO_ORDO ESP=X UPR=IU_DT2 MU=S01
```

libère la réservation faite sur la ressource PRIO_ORDO par la tâche définie par l'uproc IU_DT2, l'unité de gestion S01 dans l'espace exploitation.

Les tâches

Les commandes

uxhld TSK : suspend une tâche.

uxrls TSK : réactive une tâche.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
MODEL | NOMODEL		NOMODEL

Exemple :

```
uxhld TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART
uxrls TSK EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART
```

uxlst TSK : liste les tâches.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	*
VSES=	3n *	*
UPR=	10c *	*
VUPR=	3n *	*
MU=	10c *	*

MODEL | NOMODEL FULL OUTPUT=file_name		NOMODEL INF

Exemple :

```
uxlst TSK FULL OUTPUT=c:\temp\LstTsk.txt
```

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée sauf pour **uxadd**) **obligatoire**.

VUPR : Version d'uproc. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une uproc.

Par défaut, toutes les versions des uprocs renseignées sont considérées (inutile avec SIM et EXP).

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée sauf pour **uxadd**).

VSES : Version de session. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une session.

Par défaut, toutes les versions des sessions renseignées sont considérées (inutile avec EXP).

MU : Code du modèle ou de l'unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**), saisie **obligatoire**.

NOMODEL : nature de la tâche.

Indique que la tâche sélectionnée est de nature non modèle.

MODEL indique que la tâche sélectionnée est de nature modèle.

FULL : Option d'affichage. Affiche toutes les informations disponibles sur la liste des tâches.

Autre valeur (par défaut) INF : affiche uniquement l'entête de la tâche.

FULL et INF sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items.

Exemples

```
uxhld TSK EXP UPR=D_LOAD_FIL SES=D_LOAD_BCK MU=SIEGE NOMODEL
```

désactive la tâche non modèle de la session D_LOAD_BCK dont l'uproc d'en-tête est D_LOAD_FIL sur l'unité de gestion SIEGE dans l'espace exploitation.

```
uxlst TSK EXP UPR=D_LOAD_FIL SES=D_LOAD_BCK MU=SIEGE FULL
```

liste l'ensemble des caractéristiques de la tâche qui vient d'être désactivée.

```
uxrls TSK EXP UPR=D_LOAD_FIL SES=D_LOAD_BCK MU=SIEGE
```

réactive la tâche de l'uproc IU_TEST sur l'unité de gestion SIEGE dans l'espace exploitation.

Le suivi d'exploitation

Les lancements prévus

Les commandes

uxadd FLA : crée un lancement prévu dans la liste des lancements.

La commande affiche dans la sortie standard le numéro de lancement associé avec le format suivant : "numlanc nnnnnnn" (cf. manuel d'administration pour désactiver cet affichage).

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c	" "
VSES=	3n	
UPR=	**10c**	
VUPR=	3n	
MU=	**10c**	
NSEQ=	7n	0000000
PRINTER=	4c	
PRTY=	3n	100
QUEUE=	20c	SYS_BATCH
USER=	**12c**	
LSTART=	(U_FMT_DATE, hhmm)	Heure système
LEND=	(U_FMT_DATE, hhmm)	Heure système + 5 minutes
EXLW=	(hhmm,hhmm)	(0000,0000)
PDATE=	U_FMT_DATE	
LAUNCHVAR=	((var,val),…)	
FORCE | NOFORCE		NOFORCE
CENTRAL | NOCENTRAL		NOCENTRAL
RESTART | NORESTART		NORESTART
BYPASS | NOBYPASS		NOBYPASS

Exemple :

```
uxadd FLA UPR=D_LOAD_FIL MU=HEADQUART USER=sup500e
LSTART="(20020901,2000)" LEND="(20020901,2200)" PDATE=20020901
LAUNCHVAR="((VARDATE,2010/01/01))"
```

uxupd FLA : modifie un lancement prévu.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "

VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
NSEQ=	**7n ***	
SINCE=	(U_FMT_DATE, hhmm)	
BEFORE=	(U_FMT_DATE, hhmm)	
STATUS=	L | H | W | O | R	
PRINTER=	4c	
PRTY=	3n	
QUEUE=	20c	
USER=	12c	
LSTART=	(U_FMT_DATE, hhmm)	
LEND=	(U_FMT_DATE, hhmm)	
EXLW=	(hhmm,hhmm)	
NOW		Heure système, Heure système + 5 minutes
PDATE=	U_FMT_DATE	
LAUNCHVAR=	((var,val),…)	
FORCE | NOFORCE		
CENTRAL | NOCENTRAL		
RESTART | NORESTART		
BYPASS | NOBYPASS		

Exemple :

```
uxupd FLA UPR=D_LOAD_FIL MU=HEADQUART LAUNCHVAR="((VARNUM,693))"
NSEQ=0000175
```

uxdlt FLA : supprime un lancement prévu.

uxhld FLA : suspend un lancement prévu.

uxrls FLA : réactive un lancement prévu.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
NSEQ=	**7n ***	

SINCE=	(U_FMT_DATE, hhmm)	
BEFORE=	(U_FMT_DATE, hhmm)	
STATUS=	L | H | W | O | R	

Exemple :

```
uxdlt FLA UPR=D_LOAD_FIL MU=HEADQUART NSEQ=0000175
uxhld FLA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART NSEQ=*
uxrls FLA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART NSEQ=*
```

uxshw FLA : affiche les caractéristiques d'un lancement prévu.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
UPR=	**10c ***	
VUPR=	3n *	
MU=	**10c ***	
NSEQ=	7n *	
SINCE=	(U_FMT_DATE, hhmm)	
BEFORE=	(U_FMT_DATE, hhmm)	
STATUS=	L | H | W | O | R	
OUTPUT=file_name		

Exemple :

```
uxshw FLA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART NSEQ=* |more
```

uxlst FLA : liste les lancements prévus.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	*
UPR=	10c *	*
VUPR=	3n *	*
MU=	10c *	*
SINCE=	(U_FMT_DATE, hhmm)	
BEFORE=	(U_FMT_DATE, hhmm)	
STATUS=	L | H | W | O | R	
FULL | INF		INF

OUTPUT=file_name		

Exemple :

```
uxlst FLA FULL OUTPUT=c:\temp\LstFla.txt
```

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée sauf pour **uxadd**).

VUPR : Version d'uproc. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une uproc.

Par défaut, toutes les versions des uprocs renseignées sont considérées (inutile avec SIM et EXP).

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée sauf pour **uxadd**).

VSES : Version de session. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une session.

Par défaut, toutes les versions des sessions renseignées sont considérées (inutile avec EXP).

MU : Code de l'unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**).

NSEQ : Numéro de séquence. Format : 7 caractères numériques, **obligatoire** sauf pour **uxadd**.

Indique quel numéro de lancement doit être affecté par la commande (pour le cas où il existe plusieurs lancements portant sur la même uproc session et unité de gestion.

SINCE : Début de la plage horaire d'observation. Défaut : (000000,0000)

Permet de déterminer, en fonction de leur plage horaire prévue, les lancements à observer (sont extraits tous les lancements dont la plage horaire présente une intersection non vide avec la plage d'observation).

Format : (U_FMT_DATE,HHMM) le format de la date est indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

BEFORE : Fin de la plage horaire d'observation. Défaut : (99991231,2359)

Format : (U_FMT_DATE,HHMM) le format de la date est indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

STATUS : Etat d'un lancement. Par défaut tous les lancements sont considérés.

Permet de sélectionner les lancements en fonction de leur état par la saisie d'une ou plusieurs des valeurs suivantes (séparées par une virgule) :

- L : attente de premier lancement,
- H : suspendu (désactivé),
- W : attente d'événement,
- O : horaire dépassé,
- R : refusé au pilotage.

PRINTER : Imprimante. Non utilisé sous UNIX, Windows et VMS

PRTY : Priorité. Format : 3 caractères numériques de 001 à 255. Valeur par défaut : 100.

Priorité d'entrée de la tâche dans la queue batch.

QUEUE : Queue batch. Format : 20 caractères alphanumériques ("$" accepté sous VMS).

Valeur par défaut : SYS_BATCH. Queue batch de soumission de la tâche.

USER : Compte de soumission. Format : 12 caractères alphanumériques.

Compte dans l'environnement duquel l'exécution sera soumise.

FORCE : Lancement forcé en fin de période. Valeur par défaut : NOFORCE.

Indique que l'exécution du lancement sera forcée en fin de plage horaire de la tâche même si toutes les conditions de l'uproc ne sont pas vérifiées.

CENTRAL : contrôle centralisé. Autre valeur : NOCENTRAL.

Valeur par défaut : indiquée par la variable S_CODCENTRAL. Indique que les différents états d'exécution de la tâche seront remontés au noeud de contrôle centralisé.

RESTART : relance automatique. Valeur par défaut : NORESTART.

Indique, dans le cas de l'utilisation d'un gestionnaire de queues batch uniquement, que le lancement sera automatiquement relancé lors du redémarrage de la machine s'il était en cours d'exécution lors de l'arrêt de celle-ci.

BYPASS : Lancement sans pilotage. Valeur par défaut : NOBYPASS.

Indique que les conditions liées à l'uproc ne seront pas examinées. L'uproc pourra donc s'exécuter même si ses conditions ne sont pas vérifiées. Dans le cas d'un lancement portant sur une session, seule la première uproc de la session bénéficie d'un lancement sans pilotage, les uprocs suivantes verront leurs conditions normalement vérifiées.

LSTART : Date et heure de début de plage horaire du lancement. Format (U_FMT_DATE,HHMM) où :

- U_FMT_DATE est la date de début du lancement au format indiqué par la valeur de la variable U_FMT_DATE,
- HHMM est l'heure de début du lancement en heures et minutes.

Le nombre de caractères doit être respecté exactement. **Obligatoire** pour **uxadd**. Incompatible avec **NOW**.

LEND : Date et heure de fin de plage horaire du lancement. Format (U_FMT_DATE,HHMM) où :

- U_FMT_DATE est la date de fin du lancement au format indiqué par la valeur de la variable U_FMT_DATE,
- HHMM est l'heure de fin du lancement en heures et minutes.

Le nombre de caractères doit être respecté exactement. **Obligatoire** pour **uxadd**. Incompatible avec **NOW**.

EXLW : Plage horaire d'exclusion du lancement. Format (HHMM,HHMM) où :

- (HHMM est l'heure de début de la plage horaire d'exclusion du lancement en heures et minutes.
- HHMM) est l'heure de fin de la plage horaire d'exclusion du lancement en heures et minutes.

Le nombre de caractères doit être respecté exactement. Incompatible avec **NOW**.

NOW : Début de la fenêtre de lancement : heure courante du noeud de l'environnement courant et durée de 5 minutes.

Peut être utilisé avec la commande **uxupd FLA** uniquement. Incompatible avec les items LSTART LEND et EXLW.

PDATE : Date de traitement associée au lancement. Format indiqué par la variable U_FMT_DATE.

La valeur dépend de la période fonctionnelle de l'uproc. Le nombre de caractères doit être respecté exactement.

LAUNCHVAR : Liste complète des variables du lancement et de leurs valeurs. Format : ((var,val),(var),…)

La liste est encadrée de parenthèses et de guillemets, chaque couple variable – valeur est encadré de parenthèses, chaque couple variable – valeur est séparé du suivant par une virgule (sans espace).

Le nom de la variable doit être choisi dans la liste des variables de l'uproc identifiant le lancement.

La valeur de la variable, si elle est indiquée remplace la valeur de la variable de la tâche ou de l'uproc, si la valeur n'est pas indiquée (la virgule ne doit pas être présente) c'est la valeur par défaut de la variable de la tâche ou de l'uproc qui est prise en compte.

Si la valeur contient des caractères non alphanumériques (espaces, /…) elle doit être encadrée de guillemets.

Le format de la valeur doit être compatible avec le format déclaré lors de la création de la variable dans l'uproc.

INF : Option d'affichage des informations. Valeur par défaut.

Permet l'affichage du nom des lancements uniquement.

Autre valeur : FULL : Permet d'accéder à l'ensemble des informations des lancements.

INF et FULL sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items.

Exemples

```
uxadd FLA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL VUPR=000 MU=SIEGE
USER=admin LSTART=(19990507,1255) LEND=(19990507,1700) PDATE=19990605
FORCE
```

crée un lancement à partir de la session D_LOAD_BCK, uproc entête D_LOAD_FIL sur l'unité de gestion SIEGE dans l'espace d'exploitation pour le compte utilisateur admin. Le lancement est prévu le 7 mai 1999 de 12h55 à 17h00 avec l'option lancement forcé en fin de période dans le cas où la variable U_FMT_DATE vaut YYYYMMDD.

```
uxupd FLA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL VUPR=000 MU=SIEGE NOFORCE
NSEQ=0000002
```

met à jour le lancement créé ci-dessus en indiquant que le lancement ne sera plus forcé en fin de période s'il était nécessaire. Le numéro de séquence NSEQ est donné par une commande **uxlst fla**.

```
uxhld FLA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL VUPR=000 MU=SIEGE
NSEQ=0000002
```

suspend le lancement créé ci-dessus.

```
uxadd FLA UPR=IU_LPR mu=X01 USER=root PDATE=00/00/0000
LSTART="(08/01/1998,1000)" LEND="(09/12/1998,1000)"
LAUNCHVAR="((VARTXT,TXT_VAL_FLA),(VARDATE,2010/01/01))"
```

crée un lancement de l'uproc IU_LPR sur l'unité de gestion X01avec de nouvelles valeurs pour les deux variables de la tâche.

```
uxupd FLA UPR=IU_LPR MU=X01 NSEQ="*" LAUNCHVAR=((VARNUM,693))
```

modifie la valeur de la variable VARNUM, les valeurs de VARTXT et VARDATE sont conservées.

Les événements

Les commandes

uxadd EVT : ajoute un événement dans la liste des événements.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c	" "
VSES=	3n	
NSESS=	7n	
UPR=	**10c**	
VUPR=	3n	
MU=	**10c**	
USER=	**12c**	
STEP=	2n	00
PDATE=	U_FMT_DATE	
NSTAT=S | P | E | F | T | A	1c	T

Exemple :

```
uxadd EVT EXP SES=D_LOAD_BCK NSESS=0000010 UPR=D_LOAD_FIL MU=HEADQUART
USER=univ50e PDATE=20021030 NSTAT=E
```

uxupd EVT : modifie un événement.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
NSESS=	7n	
UPR=	**10c ***	
VUPR=	3n *	
NUPR=	7n	
MU=	**10c ***	
USER=	12c	
STEP=	2n	
PDATE=	U_FMT_DATE	
NSTAT=S | P | E | F | T | A	1c	

Exemple :

```
uxupd EVT EXP SES=\" \" UPR=D_LOAD_FIL MU=HEADQUART NUPR=0000118
NSTAT=A
```

uxshw EVT : affiche les caractéristiques d'un événement.

Item	Format	Défaut
SOC=	6c	S_COMPANY

NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
NSESS=	7n	
UPR=	**10c ***	
VUPR=	3n *	
NUPR=	7n	
MU=	**10c ***	
USER=	12c	
STEP=	2n	
PDATE=	U_FMT_DATE	
STATUS=S | P | E | F | T | A	1c *	
OUTPUT=file_name		

Exemple :

```
uxshw EVT EXP SES=* UPR=* MU=HEADQUART
```

uxdlt EVT : supprime un événement.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
NSESS=	7n	
UPR=	**10c ***	
VUPR=	3n *	
NUPR=	7n	
MU=	**10c ***	
USER=	**12c ***	
STEP=	2n	
PDATE=	**U_FMT_DATE**	
NSTAT=S | P | E | F | T | A	1c	

Exemple :

```
uxdlt EVT EXP SES=\" \" UPR=D_LOAD_FIL MU=HEADQUART PDATE=20020831
USER=admin
```

uxlst EVT : liste les événements.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	
VSES=	3n *	
NSESS=	7n	

UPR=	10c *	
VUPR=	3n *	
NUPR=	7n	
MU=	10c *	
SINCE=	(U_FMT_DATE, hhmm)	
BEFORE=	(U_FMT_DATE, hhmm)	
STATUS=S | P | E | F | T | A	1c *	
FULL | INF		INF
OUTPUT=file_name		

Exemple :

```
uxlst EVT FULL OUTPUT=c:\temp\LstEvt.txt
```

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée sauf pour **uxadd**).

VUPR : Version de l'uproc. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une uproc.

Par défaut, toutes les versions des uprocs renseignées sont considérées (inutile avec SIM et EXP).

NUPR : Numéro d'exécution de l'uproc. Format : 7 caractères numériques.

Permet la saisie du numéro d'exécution de l'uproc. Ce numéro permet de différencier deux événements portant sur la même uproc au sein de la même session sur la même unité de gestion pour la même date de traitement.

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée sauf pour **uxadd**).

VSES : Version de session. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une session.

Par défaut, toutes les versions des sessions renseignées sont considérées (inutile avec EXP).

NSESS : Numéro d'exécution de la session. Format : 7 caractères numériques.

Permet la saisie du numéro d'exécution de la session. Ce numéro permet de différencier deux événements portant sur la même session sur la même unité de gestion pour la même date de traitement.

MU : Code de l'unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**) **obligatoire**.

SINCE : Début de la plage horaire d'observation. Défaut : (000000,0000).

Permet de déterminer, en fonction de leur date et heure d'exécution, les événements à observer.

Format : (U_FMT_DATE,HHMM) la date au format indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

BEFORE : Fin de la plage horaire d'observation. Défaut : (9991231,2359)

Format : (U_FMT_DATE,HHMM) la date au format indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

STATUS : Etat d'un événement. Par défaut tous les lancements sont considérés : *.

Permet de sélectionner les événements en fonction de leur état par la saisie d'une ou plusieurs des valeurs suivantes (séparées par une virgule) :

- S : pilotage en cours (Scheduling),
- P : attente exécution (Pending),
- E : exécution en cours,
- F : consignes en cours,
- T : terminé,
- A : incidenté (Aborted).

USER : Compte de soumission. Format : 12 caractères alphanumériques.

Compte de soumission pour lequel l'événement est utilisé.

NSTAT : Etat de l'événement. Indique l'état dans lequel l'événement doit être créé, modifié ou supprimé.

Format : 1 caractère alphabétique parmi :

- * : quelque que soit l'état,
- S : pilotage en cours (Scheduling),
- P : attente exécution (Pending),
- E : exécution en cours,
- F : consignes en cours,
- T : terminé,
- A : incidenté (Aborted).

STEP : numéro de jalon de l'événement. Format : 2 caractères numériques.

Indique le numéro de jalon associé à l'événement.

PDATE : Date de traitement associée à l'événement. Le nombre de caractères doit être respecté exactement.

Format indiqué par la valeur de la variable U_FMT_DATE. La valeur dépend de la période fonctionnelle de l'uproc.

INF : Option d'affichage des informations. Valeur par défaut.

Permet l'affichage du nom des événements uniquement.

Autre valeur : FULL : Permet d'accéder à l'ensemble des informations des événements.

INF et FULL sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items.

Exemples

```
uxadd EVT EXP UPR=D_LOAD_FIL MU=SIEGE USER=admin PDATE=19990507
NSTAT=T
```

crée un événement à partir de l'uproc D_LOAD_FIL sur l'unité de gestion SIEGE dans l'espace d'exploitation pour le compte utilisateur admin dans l'état terminé pour une date au format YYYYMMDD.

```
uxupd EVT EXP SES=\" \" UPR=D_LOAD_FIL MU=SIEGE NSTAT=A
```

modifie l'état de l'événement créé ci-dessus en indiquant qu'il se trouve désormais dans l'état incidenté. L'item SES permet ici de spécifier que le nom de session est à blanc.

```
uxadd EVT EXP SES=D_LOAD_BCK NSESS=0000001 UPR=D_LOAD_FIL MU=SIEGE
USER=admin PDATE=19990507 NSTAT=A
```

crée un événement à partir de la session D_LOAD_BCK, de l'uproc D_LOAD_FIL sur l'unité de gestion SIEGE dans l'espace d'exploitation pour le compte utilisateur admin dans l'état incidenté pour une date au format YYYYMMDD.

```
uxdlt EVT EXP SES=\" \" UPR=D_LOAD_FIL MU=SIEGE USER=* STATUS=A
SINCE=(19990507,0000) BEFORE=(19990508,0000)
```

supprime tous les événements de l'uproc D_LOAD_FIL sans session SES_TEST créés en espace exploitation entre le 7 mai 1999 et le 8 mai 1999 dans l'état incidenté pour une date au format YYYYMMDD.

Les exécutions

Les commandes

uxpur CTL : purge des exécutions de la liste des exécutions.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
NSESS=	7n *	
UPR=	**10c ***	
VUPR=	3n *	
NUPR=	7n *	
MU=	**10c ***	
SINCE=	(U_FMT_DATE,hhmm)	
BEFORE=	(U_FMT_DATE,hhmm)	
STATUS=	S | P | E | F | T | A | *	
OUTPUT=file_name		

Exemple :

```
uxpur CTL EXP SES=* UPR=* MU=* SINCE=(20020101,0000)
BEFORE=(20020701,0000)
```

uxend CTL : arrête une exécution en cours (état "E") et par conséquent tous les process qui y sont rattachés. L'état est alors positionné à Incidenté ("A").

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
NSESS=	7n *	
UPR=	**10c ***	
VUPR=	3n *	

NUPR=	7n *	
MU=	**10c ***	
SINCE=	(U_FMT_DATE,hhmm)	
BEFORE=	(U_FMT_DATE,hhmm)	
STATUS=	H O R S W P E F T A *	E
OUTPUT=file_name		

Exemple :

```
uxend CTL EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART
```

uxrst CTL : reprend une exécution.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
NSESS=	7n *	
UPR=	**10c ***	
VUPR=	3n *	
NUPR=	7n *	
MU=	**10c ***	
SINCE=	(U_FMT_DATE,hhmm)	
BEFORE=	(U_FMT_DATE,hhmm)	
STATUS=	A	
LW=	(hhmm,hhmm)	(0000,0000)
STEP=	2n	00
LAUNCHVAR=	((var,val),…)	
BYPASS | NOBYPASS		NOBYPASS
OUTPUT=file_name		

Exemple :

```
uxrst CTL EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART
LW=(1600,1700) BYPASS LAUNCHVAR=((VARTXT,O),(VARNUM,200))
```

uxlst CTL : liste les exécutions de façon plus ou moins complète en fonction des critères de sélection et des informations demandées.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
VSES=	3n *	
NSES=	7n *	
UPR=	**10c ***	
VUPR=	3n *	
NUPR=	7n *	

MU=	**10c ***	
SINCE=	(U_FMT_DATE,hhmm)	
BEFORE=	(U_FMT_DATE,hhmm)	
NOTIFY		
STATUS=	H O R S W P E F T A *	*
INF | HST | LOG | FULL | MGT | VAR		INF
OUTPUT=file_name		

Exemple :

```
uxlst CTL EXP FULL OUTPUT=c:\temp\LstCtl.txt
```

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée sauf pour **uxrst**).

VUPR : Version d'uproc. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une uproc.

Par défaut, toutes les versions des uprocs renseignées sont considérées (inutile avec SIM et EXP).

NUPR : Numéro d'exécution de l'uproc. Format : 7 caractères numériques.

Permet la saisie du numéro d'exécution de l'uproc. Ce numéro permet de différencier deux événements portant sur la même uproc au sein de la même session sur la même unité de gestion pour la même date de traitement.

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée sauf pour **uxrst**).

VSES : Version de session. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une session.

Par défaut, toutes les versions des sessions renseignées sont considérées (inutile avec EXP).

NSESS : Numéro d'exécution de la session. Format : 7 caractères numériques.

Permet la saisie du numéro d'exécution de la session. Ce numéro permet de différencier deux événements portant sur la même session sur la même unité de gestion pour la même date de traitement.

MU : Code de l'unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxrst**).

SINCE : Début de la plage horaire d'observation. Défaut : (000000,0000)

Permet de déterminer, en fonction de leur date de début d'exécution, les exécutions à observer.

Format : (U_FMT_DATE,HHMM) la date est au format indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

BEFORE : Fin de la plage horaire d'observation. Défaut : (19991231,2359)

Permet de déterminer, en fonction de leur date de fin d'exécution, les exécutions à observer.

Format : (U_FMT_DATE,HHMM) la date est au format indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

NOTIFY : option de sélection sur la date. Valeur par défaut " ".

Associé à SINCE et / ou BEFORE, permet de filtrer les exécutions selon leur date de dernière mise à jour par l'automate (par exemple passage de l'état "exécution en cours" à "terminé").

Si le mot clé n'est pas cité, les enregistrements sont sélectionnés selon leur date de début et de fin d'exécution.

STATUS : Etat d'une exécution. Par défaut toutes les exécutions sont considérées.

Permet de sélectionner les exécutions en fonction de leur état par la saisie d'une ou plusieurs des valeurs suivantes (séparées par une virgule) :

- H : suspendu (désactivé),
- O : horaire dépassé,
- R : refusé au pilotage.
- S : pilotage en cours (Scheduling),
- W : attente d'événement (Waiting),
- P : attente exécution (Pending),
- E : exécution en cours,
- F : consignes en cours,
- T : terminé,
- A : incidenté (Aborted).
- * : tous les états précédents y compris les reprises.

USER : Compte de soumission d'une exécution. Format : 12 caractères alphanumériques.

Permet de sélectionner les exécutions en fonction du compte de soumission de celles ci. Par défaut tous les comptes de soumission sont considérées.

LW : Plage horaire de la reprise. Format : (HHMM,HHMM) où :

- (HHMM est l'heure de début de la plage horaire de la reprise en heures et minutes.
- HHMM) est l'heure de fin de la plage horaire de la reprise en heures et minutes.

Le nombre de caractères doit être respecté exactement.

STEP : Numéro de jalon de la reprise. Format : 2 caractères numériques.

Indique le numéro de jalon à partir duquel la reprise sera effectuée.

LAUNCHVAR : Liste complète des variables de l'exécution et de leurs valeurs. Format : ((var,val),(var)…)

La liste est encadrée de parenthèses et de guillemets, chaque couple variable – valeur est encadré de parenthèses, chaque couple variable – valeur est séparé du suivant par une virgule (sans espace).

Le nom de la variable doit être choisi dans la liste des variables du lancement.

La valeur indiquée remplace la valeur de la variable dans le lancement, si la valeur n'est pas indiquée (la virgule ne doit pas être présente) c'est la valeur par défaut de la variable du lancement, de la tâche ou de l'uproc qui est prise en compte.

Si la valeur contient des caractères non alphanumériques (espaces, /…) elle doit être encadrée de guillemets.

Le format de la valeur doit être compatible avec le format déclaré lors de la création de la variable dans l'uproc.

BYPASS : Lancement sans pilotage. Valeur par défaut : NOBYPASS.

Indique que les conditions liées à l'uproc ne seront pas examinées. L'uproc pourra donc s'exécuter même si ses conditions ne sont pas vérifiées. Dans le cas d'un lancement portant sur une session, seule la

première uproc de la session bénéficie d'un lancement sans pilotage, les uprocs suivantes verront leurs conditions normalement vérifiées.

INF : Option d'affichage. Valeur par défaut.

Permet l'affichage du nom et de l'état uniquement des exécutions du suivi de l'exploitation.

Autre valeur : HST : permet l'affichage des traces automates de toutes les exécutions sélectionnées.

Autre valeur : LOG : permet l'affichage du fichier log de toutes les exécutions sélectionnées.

Autre valeur : FULL : permet l'affichage de l'ensemble des informations des exécutions,

Autre valeur : MGT : permet l'affichage des critères de gestion des exécutions,

Autre valeur : VAR : permet l'affichage de la valeur des variables de l'exécution.

INF, FULL, LOG, HST, MGT et VAR sont incompatibles entre eux.

LOG est incompatible avec OUTPUT.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items. OUTPUT est incompatible avec LOG.

Exemples

```
uxrst CTL EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=SIEGE LW=(1600,1700)
BYPASS
```

reprend l'exécution, dans l'espace exploitation, de la session D_LOAD_BCK à partir de l'uproc D_LOAD_FIL sur l'unité de gestion SIEGE sans examiner les conditions liées à cette uproc, entre 12 heures et 13 heures. Cette reprise est possible car l'événement correspondant existe dans l'état Incidenté.

```
uxlst CTL EXP SES=D_LOAD_BCK MU=SIEGE INF
```

affiche le statut de toutes les exécutions de la session D_LOAD_BCK sur l'unité de gestion SIEGE du contrôle de l'exploitation de l'espace exploitation.

```
uxlst CTL UPR=IU* SINCE=(2001/06/12,1200) BEFORE=(2001/06/13,1200)
NOTIFY
```

extrait toutes les Exécutions mises à jour (ayant changé d'état) entre le 2001/06/12 à midi et le 2001/06/13 à midi - pour une date au format YYYY/MM/DD.

```
uxpur CTL EXP UPR=D_* MU=SIEGE SINCE=(19990501,1200)
BEFORE=(19990508,1200)
```

purge le contrôle de l'exploitation de toutes les exécutions de l'uproc IU_TEST version 001 sur l'unité de gestion SIEGE entre le 1er mai 1999 à 12 heures et le 8 mai 1999 à 12 heures pour une date au format YYYYMMDD.

```
uxrst CTL UPR=IU_LPR MU=X01 NUPR=%NUMPROC%
LAUNCHVAR="((VARTXT,O),(VARNUM,200))"
```

modifie deux des valeurs des variables du lancement lors de la commande de reprise de l'exécution. les autres variables restent telles qu'elles ont été définies pour le lancement.

Les outils de contrôle

Les automates

Les commandes

uxend ATM : arrête les automates.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
CAL | LAN | ECH | SURV | ALL		ALL

Exemple :

```
uxend ATM EXP
```

uxstr ATM : démarre les automates et affecte le cycle de l'échangeur.

uxwak ATM : réveille les automates. Cette commande peut être utilisée pour réveiller un automate d'une machine distante.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
CAL | LAN | ECH | SURV | ALL		ALL
CYCLE=	6n	120

Exemple :

```
uxstr ATM EXP LAN
uxwak ATM SOC=UNIV50 NODE=saturn EXP LAN
```

uxlst ATM : affiche l'état des automates et les informations liées aux dates et heures d'arrêt, de démarrage et de réveil.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
CAL | LAN | ECH | SURV | ALL		ALL
OUTPUT=file_name		

Exemple :

```
uxlst ATM EXP OUTPUT=c:\temp\LstAtmExp.txt
```

Description des items

CAL : Permet la sélection de l'automate calculateur.

Choix incompatible avec ALL, LAN, ECH et SURV.

LAN : Permet la sélection de l'automate lanceur.

Choix incompatible avec ALL, CAL, ECH et SURV.

ECH : Permet la sélection de l'automate échangeur.

Choix incompatible avec ALL, CAL, LAN et SURV.

SURV : Permet la sélection de l'automate surveillant.

Choix incompatible avec ALL, CAL, LAN et ECH.

ALL : Permet la sélection de tous les automates.

Choix incompatible avec CAL, LAN, ECH et SURV.

CYCLE : Cycle de l'échangeur. Format : 6 caractères numériques. 120 secondes par défaut.

Indique, en secondes, la période à laquelle l'automate échangeur va tester le rétablissement de la communication avec le noeud pour lequel un échange est resté sans réponse.

La modification de la valeur de ce cycle n'est possible que par la commande **uxstr**.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items.

Exemples

`uxend ATM EXP ALL`

arrête tous les automates de l'espace exploitation.

`uxstr ATM EXP CAL`

démarre l'automate calculateur de l'espace exploitation.

`uxlst ATM EXP`

affiche les informations de tous les automates de l'espace exploitation.

`uxwak ATM SOC=UNIV43 NODE=Asterix EXP LAN`

réveille l'automate lanceur de la société UNIV43 sur le noeud Asterix dans l'espace exploitation.

Commandes "Cluster-wide" sous VMS

Les commandes UXLST ATM, UXSTR ATM et UXEND ATM sont "CLUSTER-WIDE". Elles permettent donc de manipuler les automates des autres machines du cluster.

Pour que cette fonctionnalité soit opérationnelle, il est nécessaire que certaines conditions soient remplies. Se reporter au manuel d'administration paragraphe "Architectures cluster sous VMS " pour plus le détail de ces conditions.

L'historique des exécutions

Les commandes

uxshw HCX : affiche une exécution.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
NSESS=	7n *	" "
UPR=	**10c ***	
NUPR=	7n *	*

MU=	**10c ***	
STATUS=	S | P | E | F | T | A | H | O | R | W | *	*
USER=	12c *	*
PDATE=	U_FMT_DATE	
SINCE=	(U_FMT_DATE,hhmm)	
BEFORE=	(U_FMT_DATE,hhmm)	
HIST | TECH | PLPR | VAR		

Exemple :

```
uxshw HCX EXP UPR=IU_TEST NUPR=* MU=* TECH
```

uxlst HCX : liste les exécutions de façon plus ou moins complète en fonction des critères de sélection et des informations demandées.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	*
NSESS=	7n *	*
UPR=	10c *	*
NUPR=	7n *	*
MU=	10c *	*
STATUS=	S | P | E | F | T | A | H | O | R | W | *	*
USER=	12c *	*
PDATE=	U_FMT_DATE	
SINCE=	(U_FMT_DATE,hhmm)	
BEFORE=	(U_FMT_DATE,hhmm)	
OUTPUT=file_name		

Exemple :

```
uxlst HCX EXP STATUS=A OUTPUT=c:\temp\LstCtl.txt
```

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée). Obligatoire pour **uxshw**.

NUPR : Numéro d'exécution de l'uproc. Format : 7 caractères numériques.

Permet la saisie du numéro d'exécution de l'uproc. Ce numéro permet de différencier deux exécutions portant sur la même uproc au sein de la même session sur la même unité de gestion pour la même date de traitement.

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée). Valeur par défaut " ".

NSESS : Numéro d'exécution de la session. Format : 7 caractères numériques.

Permet la saisie du numéro d'exécution de la session. Ce numéro permet de différencier deux exécutions portant sur la même session sur la même unité de gestion pour la même date de traitement.

MU : Code de l'unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée). Obligatoire pour **uxshw**.

STATUS : Etat d'une exécution. Par défaut toutes les exécutions sont considérées.

Permet de sélectionner les exécutions en fonction de leur état par la saisie d'une ou plusieurs des valeurs suivantes (séparées par une virgule) :

- S : pilotage en cours (Scheduling),
- P : attente exécution (Pending),
- E : exécution en cours,
- F : consignes en cours,
- T : terminé,
- A : incidenté (Aborted).
- H : suspendu (désactivé),
- O : horaire dépassé,
- R : refusé au pilotage.
- W : attente d'événement (Waiting),
- * : tous les états précédents y compris les reprises.

USER : Compte de soumission d'une exécution. Format : 12 caractères alphanumériques.

Permet de sélectionner les exécutions en fonction du compte de soumission de celles ci. Par défaut tous les comptes de soumission sont considérées.

PDATE : Date de traitement associée à l'exécution. Le nombre de caractères doit être respecté exactement.

Format indiqué par la valeur de la variable U_FMT_DATE. La valeur dépend de la période fonctionnelle de l'uproc.

SINCE : Début de la plage horaire d'observation. Défaut : (000000,0000)

Permet de déterminer, en fonction de leur date de début d'exécution, les exécutions à observer.

Format : (U_FMT_DATE,HHMM) la date est au format indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

BEFORE : Fin de la plage horaire d'observation. Défaut : (19991231,2359)

Permet de déterminer, en fonction de leur date de fin d'exécution, les exécutions à observer.

Format : (U_FMT_DATE,HHMM) la date est au format indiqué par la valeur de la variable U_FMT_DATE, heure et minutes sur 4 caractères. Le nombre de caractères doit être respecté exactement.

HIST | TECH | PLPR | VAR : Option d'affichage. **Obligatoire**.

HIST : permet l'affichage des traces automates de toutes les exécutions sélectionnées.

Autre valeur : TECH : permet l'affichage des informations techniques de l'exécution.

Autre valeur : PLPR : permet l'affichage des informations de lancement,

Autre valeur : VAR : permet l'affichage de la valeur des variables de l'exécution.

HIST, TECH, PLPR et VAR sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

Exemples

```
uxshw HCX EXP UPR=D_LOAD_FIL NUPR=* SES=D_LOAD_BCK NSESS=* MU=SIEGE
VAR
```

affiche les variables de l'uproc D_LOAD_FIL s'exécutant pour la session D_LOAD_BCK et l'unité de gestion SIEGE.

```
uxlst HCX SINCE="(06/01/2004,0000)" BEFORE="(06/30/2004,0000)"
```

liste l'historique des exécutions de l'espace par défaut entre le 1er et le 30 juin 2004.

Le planning prévisionnel

Les commandes

uxlst PRV : calcule le planning prévisionnel sur la période indiquée.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	
UPR=	**10c ***	
MU=	**10c ***	
DSINCE=	**U_FMT_DATE**	
DBEFORE=	**U_FMT_DATE**	
TSINCE=	**hhmmss**	
TBEFORE=	**hhmmss**	
OUTPUT=file_name		

Exemple :

```
uxlst PRV EXP SES=D* UPR=* MU=HEADQUART DSINCE=20020701 TSINCE=000000
DBEFORE=20021231 TBEFORE=000000
```

Description des items

UPR : Code uproc. Requiert la saisie d'un code uproc (valeur générique acceptée).

Format : 10 caractères alphanumériques.

SES : Code session. Requiert la saisie d'un code session (valeur générique acceptée).

Format : 10 caractères alphanumériques.

MU : Code de l'unité de gestion. Requiert la saisie d'un code unité de gestion (valeur générique acceptée).

Format : 10 caractères alphanumériques.

DSINCE : Date de début du calcul.

Format : indiqué par la valeur de la variable U_FMT_DATE.

DBEFORE : Date de fin du calcul.

Format : indiqué par la valeur de la variable U_FMT_DATE.

TSINCE : Heure de début du calcul.

Format : 4 caractères numériques au format hhmm (heure, minute).

TBEFORE : Heure de fin du calcul.

Format : 4 caractères numériques au format hhmm (heure, minute).

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items.

Exemple

```
uxlst prv app upr=RCA_DEB dsince=01011997 dbefore=31121998
```

calcul le planning prévisionnel de l'espace application entre le 1er janvier 1997 et le 31 décembre 1998.

```
session  VER Uproc    VER MU    DATE
         000 RCA_DEB  001 ISM  02011997  02011997 1200  02011997 1300
RC       001 RCA_DEB  001 ISM  02011997  02011997 1200  02011997 1300
RC_TEST  001 RCA_DEB  001 ISM  02011997  02011997 1200  02011997 1300
         000 RCA_DEB  001 ISM  02011998  02011998 1200  02011998 1300
RC       001 RCA_DEB  001 ISM  02011998  02011998 1200  02011998 1300
RC_TEST  001 RCA_DEB  001 ISM  02011998  02011998 1200  02011998 1300
```

La première date est la date de traitement affectée au lancement.

La seconde date / heure est la date et heure de début de plage horaire du lancement.

La troisième date / heure est la date et l'heure de fin de plage horaire du lancement.

Les statistiques

Les commandes

uxadd STA : ajoute une tâche à la liste des statistiques. Cette commande ne crée que l'enveloppe des statistiques, les autres informations doivent être saisies par l'intermédiaire de la commande **uxupd**.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c	" "
UPR=	**10c**	
MU=	**10c**	

Exemple :

```
uxadd STA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART
```

uxupd STA : modifie les statistiques d'une tâche.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
UPR=	**10c ***	
MU=	**10c ***	

STATIST add : ajoute des statistiques.

Item	Format	Défaut

NBEXEC=	2n [0..99]	
CPU=	9n	0
ELAPS=	9n	0
PGF=	9n	0

Exemple :

```
uxupd STA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART STATIST add
NBEXEC=10 CPU= ELAPS=
```

STATIST upd : modifie des statistiques.

Item	Format	Défaut
NBEXEC=	2n [0..99]	
CPU=	9n	0
ELAPS=	9n	0
PGF=	9n	0

Exemple :

```
uxupd STA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART STATIST upd
NBEXEC=10 CPU=61 ELAPS=615
```

STATIST dlt : supprime des statistiques.

Item	Format	Défaut
NBEXEC=	2n [0..99]	
CPU=	9n	
ELAPS=	9n	
PGF=	9n	

Exemple :

```
uxupd STA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART STATIST dlt
NBEXEC=10 ELAPS=615
```

uxshw STA : affiche les statistiques d'une tâche.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	" "
UPR=	**10c ***	
MU=	**10c ***	
INF | FULL		INF
OUTPUT=file_name		

Exemple :

```
uxshw STA EXP SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART FULL
```

uxdlt STA : supprime une tâche de la liste des statistiques.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE

SES=	10c *	" "
UPR=	**10c ***	
MU=	**10c ***	

Exemple :

```
uxdlt STA SES=D_LOAD_BCK UPR=D_LOAD_FIL MU=HEADQUART
```

uxlst STA : liste les statistiques.

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
EXP | SIM | INT | APP	3c	S_ESPEXE
SES=	10c *	*
UPR=	10c *	*
MU=	10c *	*
FULL | INF		FULL
OUTPUT=file_name		

Exemple :

```
uxlst STA FULL OUTPUT=c:\temp\LstSta.txt
```

Description des familles

Une seule des familles décrites ci-dessous peut être utilisée dans une commande uxupd.

STATIST add : ajoute les statistiques d'une exécution à une tâche : temps CPU, elapsed ou page faults.

STATIST upd : modifie les statistiques d'une exécution : temps CPU, elapsed ou page faults.

STATIST dlt : supprime une exécution des statistiques: temps CPU, elapsed ou page faults.

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc (valeur générique acceptée sauf pour **uxadd**) **obligatoire**.

SES : Code session. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code session (valeur générique acceptée sauf pour **uxadd**).

MU : Code de l'unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée sauf pour **uxadd**) **obligatoire**.

NBEXEC : numéro de l'exécution. Sur 2 caractères numériques, il doit être compris entre 0 et 99.

Pour l'exécution de la commande "uxupd … STATIST add …", ce numéro doit être immédiatement supérieur au denier numéro d'exécution enregistré pour cette tâche, sinon l'exécution de la commande est refusé. Les numéros d'exécution existant des statistiques d'une tâche peuvent être affiché par une commande "uxshw STA … FULL".

CPU : temps CPU d'une exécution exprimé en secondes. Sur 9 caractères numériques.

ELAPS : temps ELAPS d'une exécution exprimé en secondes. Sur 9 caractères numériques.

PGF : page faults d'une exécution (sur VMS uniquement). Sur 9 caractères numériques.

INF : Option d'affichage des informations. Valeur par défaut de la commande uxshw.

Utilisé avec la commande uxlst permet l'affichage du temps cpu uniquement.

Utilisé avec la commande uxshw permet l'affichage des moyennes.

INF et FULL sont incompatibles entre eux.

FULL : Option d'affichage des informations. Valeur par défaut de la commande uxlst.

Utilisé avec la commande uxlst permet l'affichage de toutes les informations : session, uproc, UG, temps CPU, temps elapsed et page Faults.

Utilisé avec la commande uxshw permet l'affichage des moyennes et du détail des temps pour chaque exécution.

INF et FULL sont incompatibles entre eux.

OUTPUT : Stockage des résultats. Si l'item n'est pas renseigné la sortie se fait à l'écran.

La description de cet item est fournie dans le dictionnaire d'items.

Exemples

```
uxadd STA EXP UPR=D_LOAD_FIL MU=SIEGE
```

crée une entrée dans la liste des statistiques pour l'uproc D_LOAD_FIL sur l'unité de gestion SIEGE dans l'espace d'exploitation.

```
uxupd STA EXP UPR=D_LOAD_FIL MU=SIEGE STATIST add NBEXEC=1 CPU=10
ELAPS=60
```

modifie la statistique créée précédemment : créé l'exécution n°1 avec un temps CPU de 10 secondes et un temps elapsed de 60 secondes.

```
uxupd STA EXP UPR=D_LOAD_FIL MU=SIEGE STATIST upd NBEXEC=1 CPU=5
ELAPS=30
```

modifie la statistique : modifie l'exécution n°1 avec un temps CPU de 5 secondes et un temps elapsed de 30 secondes.

```
uxdlt STA EXP UPR=D_LOAD_FIL MU=SIEGE
```

supprime la tâche de l'uproc D_LOAD_FIL sur l'unité de gestion SIEGE de la liste des statistiques.

Gestion de batch

Présentation

Le déclenchement de travaux batchs par Dollar Universe peut être effectué par différents moyens :

- Une tâche planifiée sera pilotée par l'ordonnanceur lorsque sa date et son heure de planification coïncideront avec la date et l'heure système,
- Une tâche provoquée ou planifiée pourra être déclenchée par une intervention de l'opérateur qui demande la création ou la modification d'un lancement,
- Une tâche provoquée pourra être déclenchée par l'exécution de la commande **uxordre** (alias **uxord** sous VMS).

Dans tous les cas, le déclenchement de la tâche débute (lorsque la plage horaire d'exécution de la tâche est valide) par l'examen de ses conditions d'exécution lors de la phase de pilotage. Si ces conditions sont satisfaites, la tâche sera effectivement exécutée.

La durée d'exécution du batch pourra être surveillée et interrompue si elle excède le temps maximum indiqué. Cette fonction est obtenue par la commande **uxsurjob** (non disponible sous VMS).

L'objet de ce chapitre est la description des commandes : **uxordre** et **uxsurjob**.

Déclenchement de batch

Dollar Universe fournit une commande permettant de déclencher des traitements depuis l'environnement CL ou un script.

Cette commande permet de déclencher une tâche existante dans le paramétrage de Dollar Universe et définie comme étant une tâche provoquée. La tâche pourra être déclenchée avec ses paramètres de définition, ou en y substituant explicitement certains paramètres décrits ci-après.

La commande uxordre lancée sur une tâche désactivée créé désormais par défaut un lancement désactivé ; si le nom logique U_WRITE_HELD'Espace' est valorise a 'N' alors aucun lancement n'est créé (cf. manuel d'administration).

La commande **uxordre** (alias **uxord** sous VMS) permet également le passage de paramètres applicatifs (jusqu'à trente paramètres) :

- à la procédure, si la tâche est une uproc simple,
- à l'uproc entête de la session, si la tâche porte sur une session.

La commande affiche dans la sortie standard le numéro de lancement associé avec le format suivant : "numlanc nnnnnnn" (cf. manuel d'administration pour désactiver cet affichage).

Format

Le format de la commande est le suivant : **uxordre** (**uxord** sur VMS)

Item	Format	Défaut
SOC=	6c	S_COMPANY
NODE=	10c	S_NOEUD
ESP=X | S | I | A	1c	S_ESPEXE
SES=	10c	" "
VSES=	3n	
UPR=	**10c**	
VUPR=	3n	
UG=	**10c**	
AFTER=	"U_FMT_DATE-hhmm"	system date and time
ELAPS=	"(hhhmm)"	99959
EXCLU=	"(hhmm,hhmm)"	(0000,0000)
USER=	12c	schedul
QUEUE=	20c	schedul
PRINTER=	4c	schedul
DTRAIT=	U_FMT_DATE	schedul
CENTRAL | NOCENTRAL		schedul
FORCE | NOFORCE		schedul
BYPASS		
PARAM=P1,P2,…	255c	

Exemple :

```
uxordre UPR=1ST_JOB UG=SATURN PARAM=34,/tmp,type:2
```

Description des items

UPR : Code uproc. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code uproc **obligatoire.**

VUPR : Version d'uproc. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une uproc.

Par défaut, toutes les versions des uprocs renseignées sont considérées (inutile avec les espaces S et X).

SES : Code session. Format : 10 caractères alphanumériques.

Permet la saisie d'un code session.

VSES : Version de session. Format : 3 caractères numériques.

Permet la saisie d'une version particulière d'une session.

Par défaut, toutes les versions des sessions renseignées sont considérées (inutile dans l'espace X).

UG : Code unité de gestion. Format : 10 caractères alphanumériques.

Requiert la saisie d'un code unité de gestion (valeur générique acceptée) **obligatoire**.

AFTER : Date et heure de début de plage de lancement. Permet de déclencher un traitement en différé.

Défaut : les informations correspondantes dans la tâche.

Format : U_FMT_DATE-HHMM, le format de la date est indiqué par la valeur de la variable U_FMT_DATE, l'heure est sur 4 caractères.

ELAPS : Durée de la plage de lancement. Format : hhhmm.

Défaut : 99959 (999h 59m soit 41,6 jours).

EXCLU : Plage horaire d'exclusion. Format : hhmm,hhmm. Défaut : vide.

Permet d'exclure un intervalle horaire de la plage de lancement.

USER : Compte de soumission du lancement. Format : 12 caractères alphanumériques.

Défaut : l'information correspondante dans la tâche.

Doit être référencé en tant qu'utilisateur dans les tables d'administration correspondantes de Dollar Universe.

QUEUE : Queue batch d'exécution du lancement. Format : 20 caractères alphanumériques.

Défaut : l'information correspondante dans la tâche.

Doit exister au niveau du système d'exploitation.

PRINTER : Imprimante attachée à la tâche. Format : nom logique à quatre caractères.

Défaut : l'information correspondante dans la tâche.

CENTRAL : Indicateur de "contrôle centralisé". Autre valeur : NOCENTRAL.

Défaut : l'information correspondante dans la tâche.

FORCE : Indicateur de "lancement forcé en fin de période". Autre valeur : NOFORCE.

Défaut : l'information correspondante dans la tâche.

BYPASS : Permet d'effectuer un lancement sans vérifier les conditions de l'uproc.

DTRAIT : Date de traitement. Format : indiqué par la valeur de la variable U_FMT_DATE.

Défaut : l'information correspondante dans la tâche.

PARAM : Paramètres applicatifs.

Permet le passage de 1 à 30 paramètres à la première uproc de la tâche. Ces paramètres peuvent être récupérés :

- dans le CL de l'uproc, sous la forme $1 à $30, ou sous la désignation Dollar Universe S_P1 à S_P30,
- dans les scripts lancés par la coquille batch : U_ANTE_UPROC et U_POST_UPROC, sous la désignation Dollar Universe.

Le nombre maximum de paramètres transmis est de 30. La longueur de chaque paramètre ne doit pas excéder 255 caractères et la longueur totale (somme des longueurs + 2 fois le nombre de paramètres) doit être inférieure à 2074 caractères.

Format : les paramètres saisis sont séparés par des virgules, chaque paramètre peut avoir un format quelconque sauf s'il comporte un espace, il faut alors l'indiquer entre guillemets.

Exemple

```
uxordre SOC=UNIX41 NODE=ds5200 ESP=X UPR=IU_DTS UG=S01
PARAM=34,/tmp,type:2
```

Cette commande déclenche la tâche provoquée "IU_DTS S01", existante dans la société UNIX41 sur le noeud ds5200 dans l'espace d'Exploitation avec ses paramètres de définition et passe à la procédure IU_DTS les paramètres suivants : 34, /tmp, type:2.

Surveillance de la durée d'exécution

La durée d'exécution d'un batch (temps elapsed) peut être surveillée par les commandes **uxsurjob, uxspvjob** ou **uxalrjob**.

- **uxsurjob** permet d'interrompre l'exécution dès lors que le temps dépasse celui indiqué en paramètre à la commande.
- **uxspvjob** permet d'exécuter un script utilisateur au delà du temps indiqué (sous Windows et Unix, précis à la minute).
- **uxalrjob** permet d'exécuter un script utilisateur au delà du temps indiqué (sous Windows, Unix et VMS, précis à la seconde).

uxsurjob

Lorsque le temps indiqué est écoulé, Dollar Universe tue le batch (kill) et tous les sous process qui en sont issus dans la mesure où ceux ci y sont rattachés.

Cette commande peut être introduite dans le pré-traitement U_ANTE_UPROC si l'on souhaite qu'aucune exécution de l'environnement de la société et du noeud ne dépasse un temps imparti, ou bien dans le script de l'uproc si la durée maximum d'exécution est spécifique pour chaque uproc ou si la surveillance ne porte que sur certaines exécutions.

Format

Le format de la commande est le suivant :

uxsurjob

Verbe	Item	
uxsurjob	**NB_MIN**	$S_NUMPROC | %S_NUMPROC%

Elle doit être lancée en arrière plan en utilisant le signe & sous UNIX.

Description des items

NB_MIN : Temps maximum d'exécution

Indique en minutes quel est le temps elapsed maximum que l'exécution ne devra pas dépasser.

S_NUMPROC : est le numéro d'exécution de l'uproc en cours d'exécution, cette variable est automatiquement renseignée par Dollar Universe.

Sous Windows : %S_NUMPROC%.

Sous UNIX : $S_NUMPROC

Exemple

```
%UXEXE%\uxsurjob 60 %S_NUMPROC%        (sous Windows)
$UXEXE/uxsurjob 60 $S_NUMPROC &        (sous UNIX)
```

Si cette commande est exécutée dans U_ANTE_UPROC, cela aura pour conséquence que toutes les exécutions dépassant une heure d'exécution seront tuées.

Si elle est introduite dans le script d'une uproc, elle aura pour conséquence de tuer l'exécution de l'uproc si celle-ci dépasse une heure de temps elapsed.

uxspvjob

Lorsque le temps indiqué est écoulé, Dollar Universe lance le script utilisateur U_SPV_UPROC situé dans le répertoire UXMGR de la société.

Cette commande peut être introduite dans le pré-traitement U_ANTE_UPROC si l'on souhaite appliquer le même traitement à toutes les uprocs qui dépassent un temps imparti, ou bien dans le script de l'uproc si la durée surveillée est spécifique pour chaque uproc ou si la surveillance ne porte que sur certaines exécutions.

Sous Windows, la commande %UXEXE%\U_SPV_UPROC doit avoir une extension .bat ou .cmd.

Format

Le format de la commande est le suivant :

Verbe	Item
uxspvjob	**NB_SEC**

Elle doit être lancée en arrière plan en utilisant le signe & sous UNIX.

Description des items

NB_SEC : Temps maximum.

Indique en secondes quel est le temps elapsed maximum au delà duquel le script devra être lancé.

Le temps indiqué est converti en minutes et arrondi au nombre de minutes supérieur. Par exemple :

- 5s 1 min
- 58s 1 min
- 67s 2 min

Exemple

```
%UXEXE%\uxspvjob 550        (sous Windows)
$UXEXE/uxspvjob 550         (sous UNIX)
```

Au delà de 10 minutes d'exécution de l'uproc, le script U_SPV_UPROC sera exécuté.

uxalrjob

Lorsque le temps indiqué est écoulé, Dollar Universe lance par défaut le script utilisateur U_ALR_JOB situé dans le répertoire UXMGR de la société.

Le nom du script peut néanmoins être nommé autrement en valorisant la variable U_ALR_JOB.

Cette commande peut être introduite dans le pré-traitement U_ANTE_UPROC si l'on souhaite appliquer le même traitement à toutes les uprocs qui dépassent un temps imparti, ou bien dans le script de l'uproc si la durée surveillée est spécifique pour chaque uproc ou si la surveillance ne porte que sur certaines exécutions.

Sous Windows, la commande %UXEXE%\U_ALR_JOB doit avoir une extension .bat ou .cmd.

Format

Le format de la commande est le suivant :

```
%UXEXE%\uxalrjob.exe N      sur Windows
$UXEXE/uxalrjob N &         sur UNIX
$ @UXEXE:uxalrjob N         sur VMS
```

Description des items

N : Temps maximum.

Indique en secondes quel est le temps elapsed maximum au delà duquel le script devra être lancé.

Exemple

```
%UXEXE%\alrjob 75    (sous Windows)
```

Au delà de 75 secondes d'exécution de l'uproc, le script U_ALR_JOB sera exécuté.

Gestion de date

Dollar Universe utilise, pour le calcul de la planification, des algorithmes perfectionnés de gestion des dates et des heures, que ce soit sur le format de la date ou sur le calcul de celle-ci, ou pour prendre en compte les différents types de jours : calendaires, ouvrés, chômés ou fériés.

Formatage et décalage de dates

La commande **uxdat** permet de modifier le format d'une date, de réaliser un décalage en jour et d'obtenir le type ou le rang d'un jour dans le calendrier.

La plus petite unité de temps traitée est le jour et la plus grande est l'année.

Les six résultats sont fournis à l'écran (Unix, Windows) ou sous la forme de six symboles (VMS). Seul, l'un des 6 résultats est renseigné selon le type demandé.

Format

Seule la cohérence syntaxique est vérifiée.

La langue par défaut est internationale (Anglais). Le choix de la langue du format est déterminé par le format lui même: j, a, s, ... pour Français et d, y, w, ... pour International.

Le résultat est en minuscules si le format l'est, il est en majuscules si au moins un caractère du format de sortie spécifié est saisi en majuscules.

Les accès aux calendriers de Dollar Universe ne se font que lors des décalages en jours ouvrés et des demandes de "Statut de la date source". Toutes les autres opérations sont indépendantes de l'environnement de Dollar Universe.

Deux syntaxes sont possibles selon le résultat que l'on veut obtenir.

Formatages et décalages

Les décalages sont traités séquentiellement de gauche à droite.

Le format de la commande est le suivant :

Verbe	Item			
uxdat	Format date source	Date source	Format date cible	Décalage

```
uxdat "jj/mm/aaaa" "01/01/1999" "jj-mmm-aa" "-1jo/1m"
```

Résultat 1 : U_DATE_RES : Date au format cible avec le décalage demandé

Type de jour

Le format de la commande est le suivant :

Verbe	Item		
uxdat	**Format date source**	**Date source**	**Type**

```
uxdat "jj/mm/aaaa" "01/01/1999" "type"
```

Le résultat dépend de la valeur prise par "type".

Résultat 2 : U_TYPE_JOUR : Statut de la date source (Ouvré, Chômé, Férié).

Résultat 3 : U_RGO_DEB_MOIS : Rang d'un jour ouvré dans le mois (début).

Résultat 4 : U_RGO_FIN_MOIS : Rang d'un jour ouvré dans le mois (fin).

Résultat 5 : U_RGO_DEB_ANNEE : Rang d'un jour ouvré dans l'année (début).

Résultat 6 : U_RGO_FIN_ANNEE : Rang d'un jour ouvré dans l'année (fin).

Description des items

Format des dates

	Format date français:	Format date internation.:
Mois sous la forme 1-12	m	m
Mois sous la forme 01-12	mm	mm
Mois sous la forme de jan à déc (jan to dec)	mmm	mmm
Mois sous la forme de janvier à décembre (january to december)	mmmm	mmmm
Jours sous la forme 1-31	j	d
Jours sous la forme 01-31	jj	dd
Jours sous la forme 1 à 7	p	p
Jours sous la forme lun à dim (mon to sun)	jjj	ddd
Jours sous la forme lundi à dimanche (monday to sunday)	jjjj	dddd
Semaine sous la forme 1-52	s	w
Semaine sous la forme 01-52	ss	ww
Quantième sous la forme 1-365	q	q
Quantième sous la forme 001-365	qqq	qqq
Années sous la forme 00 à 99	aa	yy
Années sous la forme 0000 à 9999	aaaa	yyyy

Lorsque le format d'entrée comporte une année sur 2 caractères, l'année est interprétée comme suit :

- de 00 à 40, l'année est située entre 2000 et 2040,
- de 70 à 99, l'année est située entre 1970 et 1999.

Note : la commande uxdat n'accepte toujours que des dates comprises entre 1970 et 2040.

Décalages

"n" est un entier

	Français	Internation.
Séparateur	/	/
Position	n	n
Décalage négatif	-n	-n
Décalage positif	+n	+n
Décalage positif de n jour calendaire	+nj	+nd
Décalage négatif de n jour calendaire	-nj	-nd
Décalage positif de n jour ouvré	+njo	+ndw
Décalage négatif de n jour ouvré	-njo	-ndw
Position au nième jour calendaire de la semaine	ns	nw
Position au nième jour ouvré de la semaine	nso	nww
Décalage positif de n semaine	+ns	+nw
Décalage négatif de n semaine	-ns	-nw
Position au nième jour calendaire du mois	nm	nm
Position au nième jour ouvré du mois	nmo	nmw
Position au nième jour (type p) calendaire du mois	nmp	nmp
Position au nième jour (type p) ouvré du mois	nmop	nmwp
Décalage positif de n mois	+nm	+nm
Décalage négatif de n mois	-nm	-nm
Position au nième jour calendaire de l'année	na	ny
Position au nième jour ouvré de l'année	nao	nyw
Position au nième jour (type p) de l'année	nap	nyp
Position au nième jour (type p) ouvré de l'année	naop	nywp
Position au nième mois de l'année	nma	nmy
Décalage positif de n année	+na	+ny
Décalage négatif de n année	-na	-ny

Type

Résultat :	Type Français:	Type Internation. :
Statut de la date source	ocf	ocf
Rang d'un jour ouvré dans le mois (début)	odm	wbm
Rang d'un jour ouvré dans le mois (fin)	ofm	wem
Rang d'un jour ouvré dans l'année (début)	oda	wby
Rang d'un jour ouvré dans l'année (fin)	ofa	wey

Exemples

Format

Exemples de format pour le 01/02/1999

"j"	
"jj"	01
"jjj"	lun
"jjjj"	lundi
"jjjj jj mmmm aaaa"	lundi 01 février 1999
"dd mmmm yyyy"	01 february 1999
"q"	32 (quantième : position dans l'année)
"qqq"	032 (quantième : position dans l'année)
"s"	5 (semaine)
"ss"	05 (semaine)
"jj-mmm-aaaa"	01-fev-1999
"dd-mmm-yyyy"	01-feb-1999

Décalage

Exemples de décalage pour le 01/02/1999

"+1m/1m/-1jo"	dernier jour ouvré du mois (décalage +1 mois, 1er jour du mois, décalage -1 jour ouvré)
"+1a/1a/-1s/7s"	dernier dimanche de l'année (décalage +1 an, 1er jour de l'année, décalage -1 semaine, 7ème jour)

Formatage de date et décalage d'heure

La commande **uxtim** permet de convertir une date connue ou non dans un format donné et/ou d'y ajouter un décalage horaire.

Formats

Pour une date connue, le format de la commande est le suivant :

Verbe	Item			
uxtim	**"Format date heure entrée"**	**"Date heure entrée"**	"Format date heure sortie"	"Décalage horaire"

Par exemple :

```
uxtim "hhMM ap" "0600 PM" "HH:MM" "-0h10"
```

Pour une date corespondant à la date et heure d'exécution de la commande, le format est le suivant :

Verbe	Item		
uxtim	**NOW**	"Format date heure sortie"	"Décalage horaire"

Par exemple :

```
uxtim NOW "aaaa, mmmm, jj HH:MM AP" "+2h50"
```

Items

Format date heure entrée – Format date heure sortie : respectivement le format de la date à convertir et le format souhaité.

Le format de la date est identique au format utilisé par la commande uxdat (cf. tableau ci-dessous).

Le format de l'heure utilise les symboles HH pour les heures, MM pour les minutes et AP pour indiquer le format horaire (en format 12h).

Par exemple : "15012003, 1506" correspond a "jjmmaaaa, HHMM"

	Format date français:	Format date internation.:
Mois sous la forme 1-12	m	m
Mois sous la forme 01-12	mm	mm
Mois sous la forme de jan à déc (jan to dec)	mmm	mmm
Mois sous la forme de janvier à décembre (january to december)	mmmm	mmmm
Jours sous la forme 1-31	j	d
Jours sous la forme 01-31	jj	dd
Jours sous la forme 1 à 7	p	p
Jours sous la forme lun à dim (mon to sun)	jjj	ddd
Jours sous la forme lundi à dimanche (monday to sunday)	jjjj	dddd
Semaine sous la forme 1-52	s	w
Semaine sous la forme 01-52	ss	ww
Quantième sous la forme 1-365	q	q
Quantième sous la forme 001-365	qqq	qqq
Années sous la forme 00 à 99	aa	yy
Années sous la forme 0000 à 9999	aaaa	yyyy

Date Heure entrée : date et heure à convertir selon le format indiqué en **Format date heure sortie**.

Cette date – heure doit être saisie selon le **Format date heure entrée** indiqué.

Décalage horaire : décalage à appliquer à la date – heure saisie en entrée ou à la date – heure courante si le mot clé **NOW** est indiqué.

Le format du décalage est : +/-HHHhMM :

- le signe + indique un décalage dans le futur, le signe – indique un décalage dans le passé.
- HHH est le nombre d'heure à décaler (sur 3 digits maximum),
- "h" est un délimiteur qui doit être saisi tel quel,
- MM est le nombre de minutes à décaler (entre 0 et 59).

Exemples

```
% uxtim "hhMM ap" "0600 PM" "HH:MM"
18:00
```

"0600 PM" a été converti selon le format indiqué (hhMM ap), il n'y a pas de décalage demandé.

```
% uxtim "hhMM ap" "0600 PM"
0600 PM
```

"0600 PM" n'a pas été converti, il n'y a pas de format de sortie indiqué et pas de décalage.

```
% uxtim NOW "aaaa, mmmm, jj HH:MM AP"
2003, mars, 05 03:30 PM
```

La date et heure courante a été convertie au format indiqué, pas de décalage.

```
% uxtim NOW "aaaa, mmmm, jj HH:MM AP" "+2h50"
2003, mars, 05 06:20 PM
```

La date et heure courante a été convertie au format indiqué, un décalage de 2h50 a été apliqué.

```
% uxtim "aaaa, mmmm, jj HH:MM AP" "2003, JANVIER, 21 06:26 PM"
"jj/mm/aa"
21/01/03
```

Le "21 janvier 2003 6h26 PM" a été converti au format indiqué, pas de décalage.

Le CL d'exécution

Présentation du CL

Le CL d'exécution désigne le fichier de commandes associé à l'uproc. Dans la suite de ce chapitre, toutes les procédures d'exploitation écrites en langage de commande (propre au système d'exploitation) seront désignées sous l'unique terme de "procédures".

Le CL permet d'accéder à la communication indispensable entre les procédures et Dollar Universe. Celui-ci fournit donc un ensemble de commandes qui permettent aux procédures d'acquérir les informations susceptibles de les intéresser ou de lui en transmettre.

Rappelons que les pré-traitements effectués par Dollar Universe avant le lancement effectif de l'exécution (pilotage) ne nécessitent aucune insertion de commande particulière dans le CL.

D'autre part, il est conseillé, pour des raisons de facilité, de manipulation et de maintenance de réaliser des procédures homogènes au plan fonctionnel et courtes (formules de lancement simplifiées, expression de conditions plus simples).

Les commandes proposées par Dollar Universe permettent d'obtenir des procédures parfaitement portables quels que soient le site ou la configuration utilisée, d'assurer une gestion cohérente de l'exploitation et d'exploiter dans les chaînes applicatives toute la logique induite par les concepts gérés par Dollar Universe (unité de gestion et relations entre elles, espaces, ...).

Normes d'écritures du CL

L'écriture du CL peut se faire soit sous le contrôle de Dollar Universe soit indépendamment avec l'éditeur de son choix.

Le CL peut utiliser, en dehors ou avec les commandes décrites ci-après, les variables logiques ou les variables d'environnement utilisées par Dollar Universe.

Le code retour du CL

Si Dollar Universe peut connaître la qualité technique de la terminaison d'une procédure, celle-ci peut cependant être, d'un point de vue applicatif, jugée incorrectement terminée.

Aussi et pour couvrir tous les cas possibles, Dollar Universe propose aux procédures de lui restituer un indicateur positionnant le code retour souhaité.

Deux procédures peuvent rendre un code retour : le pré traitement et le CL de l'uproc.

Le pré traitement restitue son code retour dans la variable *unix_status*, l'exit de l'uproc est restitué dans la variable *RESEXE ou S_RESEXE* sous VMS. Ils peuvent être utilisé dans le CL de l'uproc ainsi que dans le post traitement de la coquille batch.

Les commandes du CL

Les commandes spécifiques disponibles sont les suivantes :

Verbe	Thème	Fonction
uxset	**parm**	Passage de paramètres dans le cadre d'une session
uxset	**info**	Affichage d'informations supplémentaires dans le suivi des exécutions
uxset	**msg**	Ecriture de messages dans l'historique d'une uproc
uxset	**step**	Ecriture d'un jalon dans une uproc
uxset	**mgt**	Valorisation d'un critère de gestion
uxset	**var**	Reconduction de la valeur d'une variable au sein d'une session

Dans chacune de ces commandes la valeur prise par défaut des paramètres concernant les unités de gestion et les applications sont, systématiquement celles de l'environnement d'exécution.

Les commandes du CL

Les commandes décrites ci-dessous inscrivent une trace de leur exécution dans l'historique des exécutions (trace automate). Si la variable U_HISTORIC est positionnée à "N" (voir manuel d'administration), l'accès à l'historique des exécutions est impossible et une erreur sera affichée dans le log de l'uproc.

Ecriture de jalons

La commande **uxset step** permet de définir des jalons (ou steps) au sein du CL d'exécution. Ces jalons peuvent également permettre de reprendre des exécutions après un incident sans ré-exécuter systématiquement l'ensemble de l'uproc.

Les numéros de jalons sont enregistrés dans la base des événements au fur et à mesure que l'exécution de l'uproc se déroule. Les jalons permettent donc, dans le cadre du suivi dynamique d'exploitation, de suivre la progression d'exécution d'une uproc.

Dans un cas de reprise, Dollar Universe propose de saisir le numéro de jalon à partir duquel l'utilisateur désire voir démarrer son exécution de reprise (01 par défaut). La valeur du numéro de jalon saisie est disponible dans le CL de l'uproc au travers de la variable S_NUMJALON. Cette variable doit être testée dans le CL, et suivant sa valeur, le CL doit dérouter le programme à la ligne désirée.

Format

```
uxset step nn
```

nn représente un numéro de jalon sur deux caractères numériques. Le passage au jalon sera horodaté dans la trace Dollar Universe d'exécution de l'uproc.

Exemple

```
uxset step 2
```

Cette commande inscrite dans le CL de l'uproc, aura pour effet de marquer cette ligne comme étant le jalon n°2. Lors de l'exécution de la procédure, le suivi de l'exploitation sous Dollar Universe permettra de visualiser le passage à ce jalon (et aux autres).

Dans l'annexe de ce manuel, des procédures d'exemples en ksh et csh illustrent l'utilisation des jalons.

Affichage d'informations

La commande **uxset info** permet d'afficher des informations utilisateur dans la liste des exécutions.

Deux nouveaux champs sont disponibles :

- Un champ texte.
- Un nombre entier.

La valeur associée est affichée dans la liste des exécutions :

- Global Control 2.2 sous Windows : nouvelles colonnes "Informations" et "Sévérité" de la liste des exécutions,
- Interface commandes : mot clé INFO de la commande **uxlst ctl**.

Format

```
uxset info value=VALUE [severity=NUM]
```

value= saisir le texte qui doit être affiché, encadré de guillemets s'il contient des caractères 'espace'. Longueur maximum de 50 caractères.

severity= est un nombre entier de 0 à 9999. Sa saisie est facultative.

Exemple

L'exécution d'une uproc dont le script contient cette commande :

```
uxset info value="mon job est en cours d'execution" severity=0004
```

Affiche " mon job est en cours d'execution" dans la première colonne **Informations** et "0004" dans la seconde colonne **Sévérité** de la liste des exécutions du Global Control 2.2.

Génération de messages

La commande **uxset msg** permet de communiquer entre le CL d'une uproc et la *trace automate* de son exécution. Elle permet ainsi, par exemple, d'insérer des consignes d'exploitation dans cette trace en fonction du déroulement de l'exécution de la procédure.

La *trace automate* d'exécution de l'uproc est disponible dans l'interface de Dollar Universe "suivi d'exploitation" ou "historique d'exécutions" en sélectionnant cette uproc et en demandant l'option "Trace Automate" (option 10 en interface caractère).

Note : tous les messages générés par cette commande sont contenus dans les fichiers de l'historique des exécutions, celui-ci peut donc (en fonction du volume traité) prendre une taille importante.

Format

```
uxset msg "var1" "var2" ..."var8"
```

var désigne de 1 à 8 variables ou lignes de texte. Si **var** représente une variable, sa valeur sera traduite dans la trace, si **var** représente du texte elle sera transcrite à l'identique dans la trace Dollar Universe de l'exécution.

Exemple

```
uxset msg " *** UG d'execution : " "${S_CODUG}"
```

Sous UNIX, cette commande aura pour effet l'inscription du message suivant dans la trace d'exécution de l'uproc :

```
*** UG d'execution : A_PARIS
```

si **A_PARIS** est l'unité de gestion d'exécution de l'uproc.

Passage de paramètres

La commande **uxset parm** permet le passage de paramètres d'exécution entre des uprocs dans le cadre d'une SESSION. Les paramètres positionnés par une uproc pourront être utilisés par ses uprocs Filles sous la forme standard: sous les numéros 1 à 30 (sous UNIX, $1 à $30, sous Windows %1% à %30%) ou sous la forme Dollar Universe dans les variables : S_P1 à S_P30.

Les paramètres passés par cette commande sont utilisables :

- sous la forme standard : dans le script des uprocs uniquement,
- sous la forme Dollar Universe : dans le script des uprocs ou dans les scripts lancés par la coquille batch : pré traitement et post traitement.

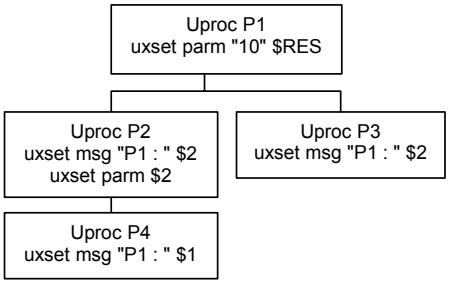
Format

```
uxset parm "parm1" "parm2" ... "parm30"
```

parm*n* désigne de 1 à 30 paramètres. Le nombre de paramètres est de 30 au maximum. D'une longueur maximale de 255 caractères chacun, la longueur totale (somme des longueurs + 2 fois le nombre de paramètres) doit être inférieure à 2074 caractères.

Exemple

Sur la SESSION suivante :



si la commande **uxset parm "10" $RES** est exécutée par l'uproc P1, l'uproc P2 pourra utiliser le second paramètre par $2 (ou S_P2), l'uproc P3 pourra utiliser les deux paramètres par $1 et $2 (ou S_P1 et S_P2).

Pour que P2 puisse transmettre les paramètres à P4, il faut exécuter dans P2 une nouvelle commande **uxset parm**.

Critères de gestion

Les critères doivent être déclarés dans le fichier de description u_ffcg42.txt (répertoire de la société). Ce fichier texte est modifiable par éditeur.

La commande **uxset mgt** permet de valoriser les critères de gestion (préalablement déclarés) au sein du script d'une uproc.

Le suivi de l'exploitation de Dollar Universe permet ensuite de positionner un filtre sur les exécutions en fonction de la valeur de ces critères dans les exécutions.

Format du fichier de référence

Le format de description des critères au sein du fichier u_ffcg42.txt est le suivant :

```
Critère:nom=valeur_nom,libelle=valeur_libellé,LG=valeur_longeur
```

où :

- **Critère** est un identifiant libre. Les critères seront pris par l'automate dans l'ordre de description indiqué dans le fichier (1ère ligne 1er critère, 2ème ligne 2ème critère..).
- **nom**, **libelle** et **LG** sont des identifiants libres.
- **valeur_nom** désigne le code du critère et sera utilisé pour la manipulation de celui-ci. Ce code doit donc être unique. Son format est de 20 caractères alphanumériques maximum, le caractère "_" est accepté.
- **valeur_libellé** désigne le texte explicatif de la fonction du critère. Ce texte apparaît dans l'écran de sélection des critères de gestion du contrôle de l'exploitation. Son format est de 40 caractères alphanumériques maximum. Les caractères "_" et " " sont acceptés.
- **longueur_max** désigne le nombre maximum de caractères qui devront être saisis lors de la valorisation du critère. Son format est de 1 à 3 caractères numériques.

Les caractères ":" "=" et "," ne peuvent pas être utilisés dans les identifiants ou les valeurs puisqu'ils sont utilisés comme délimiteurs des différents champs.

Le fichier de référence peut contenir au maximum 24 critères de gestion, la somme des longueurs des différentes valeurs des critères ne doit pas excéder 256 caractères au total.

Format de la commande

Le format de la commande est le suivant :

```
uxset mgt NOM=valeur_nom VALEUR=valeur_critère
```

où :

- **valeur_nom** désigne le nom du critère tel qu'il a été défini dans le fichier de référence.
- **valeur_critère** désigne la valeur qui est associée au critère. Le nombre maximum de caractères de cette valeur doit être inférieur ou égal à la longueur maximum qui a été définie dans le fichier de référence pour ce critère.

Si plusieurs critères doivent être valorisés dans le script d'une uproc, cette commande peut être répétée autant de fois que de critères. Chaque nouvelle valeur d'un critère venant écraser l'ancienne valeur saisie.

Exemple

Fichier de référence u_ffcg42.txt :

```
Critère1:nom=REGION,libellé=région de travail,LG=10
Critère2:nom=APPLICATION,libellé=Application de rattâchement,LG=3
Critère3:nom=CODE,libellé=code retour applicatif,LG=1
```

Ce fichier définit trois critères de gestion dont les noms sont REGION, APPLICATION et CODE.

Commandes exécutées dans le script de l'uproc:

```
uxset mgt NOM=REGION VALEUR=Ile_de_fra
uxset mgt NOM=APPLICATION VALEUR=CP
uxset mgt NOM=CODE VALEUR=4
```

la première commande valorise le critère REGION par "Ile_de_fra" d'une longueur maximum de 10 caractères, l'APPLICATION est CP et le CODE vaut 4.

Reconduction de variables

La reconduction de la valeur des variables au sein d'une session n'est pas automatique. Par défaut, si rien n'est fait, la valeur de la variable n'est pas reconduite et l'automate applique les règles de valorisation par défaut.

La reconduction de la valeur peut être demandée en exécutant la commande **uxset var** dans le C.L. de l'uproc ou dans le post-traitement (U_POST_UPROC).

La commande **uxset var** permet de reconduire la valeur des variables au sein de la session. Trois types de reconduction peuvent être demandées :

- en citant dans la commande **uxset var** les variables qui doivent être reconduites et éventuellement leur nouvelle valeur.
- de façon globale, en exécutant la commande **uxset var** sans autre complément ou bien **uxset var level ALL,** pour ne reconduire que les valeurs des variables de l'uproc mère connues de l'uproc fille (les valeurs reconduites sont les valeurs connues au début de l'exécution de l'uproc mère).
- en exécutant la commande **uxset var level INTER**, pour ne reconduire que les valeurs des variables **reçues** par l'uproc mère et connues de l'uproc fille.

La reconduction des variables au sein d'une session est assurée par l'exécution de la commande. Si l'uproc suivante dans la session n'est pas exécutée par l'automate (par exemple parce qu'elle est définie comme une tâche optionnelle), Dollar Universe transmet alors les variables à l'uproc immédiatement suivante.

Format

Les syntaxes présentées ci-dessous sont autorisées :

```
uxset var

uxset var level ALL

uxset var level INTER

uxset var "VAR1=VAL1 VAR2 VAR3=VAL3"
```

VAR désigne le nom de la variable et **VAL** la valeur associée qui sera transmise à l'uproc fille.

Si la valeur n'est pas indiquée (pas de signe égal) la dernière valeur de la variable, ou sa valeur par défaut, est transmise.

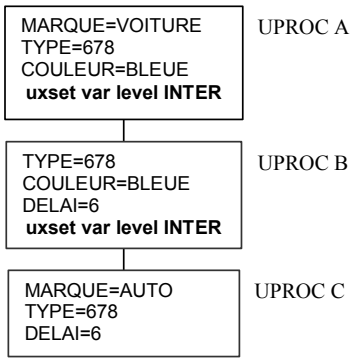
Si plusieurs variables doivent être transmises elles sont séparées par des espaces et la liste est encadrée par des guillemets.

Si une valeur contient des caractères non alphanumériques elle doit être encadrée par des guillemets. Ceux-ci doivent alors être précédés du caractère \ afin de distinguer ce guillemet de celui qui marque la fin de la liste, par exemple :

```
uxset var "VARDATE=2010/11/30 VARTXT=\"UN DEUX\" VARNUM=555"
```

Exemples

Utilisation de INTER

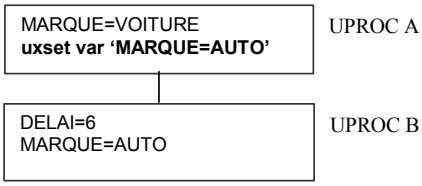


Par la commande **uxset var level INTER**, l'uproc A transmet à l'uproc B les valeurs des variables TYPE et COULEUR, la valeur de MARQUE n'est pas transmise.

Par la commande **uxset var level INTER**, l'uproc B transmet à l'uproc C les valeurs des variables TYPE et DELAI.

Si bien que l'uproc C ne connaît pas la valeur de *MARQUE* de l'uproc A et récupère la valeur par défaut.

Transmission explicite



Par la commande **uxset var 'MARQUE=AUTO'**, l'uproc A transmet à l'uproc B la nouvelle valeur de la variable *MARQUE*.

Ceci n'est possible que parce que la variable *MARQUE* est connu de l'uproc B.

Annexes

Les items des commandes

Items d'environnement

SOC	Société de travail. La Société est obligatoire; le défaut peut être généré par l'alimentation du symbole global S_SOCIETE
NODE	Noeud de travail. Le noeud par défaut est celui renseigné dans le nom logique S_NOEUD (fourni par la société)
EXP, APP…	Espace de travail. EXP est considéré par défaut (valeur par défaut de S_ESPEXE). Les espaces possibles sont APP, INT, SIM, EXP

Items d'entrée/sortie

INPUT	Fichier d'entrée. Permet la saisie d'un nom de fichier (et de sa localisation), éventuellement sous forme d'un nom logique. Utilisable uniquement avec la commande **uxins**
OUTPUT	Fichier en sortie. Permet la saisie d'un nom de fichier (et de sa localisation), éventuellement sous forme d'un nom logique, si ce fichier est obtenu par la commande **uxext** il ne pourra être utilisé que par la commande **uxins**. Le même fichier pourra être utilisé pour contenir le résultat de plusieurs commandes **uxext**. Dans ce cas les informations extraites se placeront, dans le fichier, les unes à la suite des autres, sans écraser le contenu du fichier

Items d'identification d'objets

ADM	Tables d'administration
AUT	Autorisations
APD	Répertoires applications déclarées dans les tables d'une société
APPL	Applications déclarées dans les tables d'une société
CLASS, CLASSE	Classes d'incompatibilités déclarées dans les tables d'une société
DOM	Domaines déclarés dans les tables d'une société
PROFIL	Profil utilisateur
REPERT	Répertoires d'applications ou d'unités de gestion déclarés dans les tables d'une société
RES	Référence de ressource
RUL	Code de règle

TIH, HDP	Dépendances d'unité de gestion déclarées dans les tables d'une société
TNODE	Noeuds déclarés dans les tables d'une société
TSOC	Sociétés déclarées dans les tables d'une société
TYPUG, MUT	Types d'unité de gestion déclarés dans les tables d'une société
UG MU	Code unité de gestion
UPR	Code uproc
SES	Code session
USER	Utilisateurs déclarés dans les tables d'une société
VUPR	Version d'uproc
VSES	Version de session

Autres items

ADIR	Nom du répertoire data en espace application
ADISK	Nom du disque (sous VMS) en espace application
ALNM	Nom logique (sous VMS) en espace application
ALL	Tous les objets ciblés sont visés
ATTR	Attribut d'un jour de la semaine
ATTRIB	Attribut d'une ressource
AUTHO	Masque d'autorisation dans la semaine d'une règle de planification
BEFORE	Date de fin de la période considérée ou dernière année du calendrier
BOTH	Objet autorisé dans les univers de développement et de production
BYPASS	Lancement sans pilotage
CAL	Automate calculateur
CENTRAL	Indicateur de contrôle centralisé, contraire de NOCENTRAL
CIBLE	Liste des unités de gestion cible d'une insertion pour les calendriers ou les tâches
CLEAR	Libération automatique d'une ressource
CLFIL	Fichier de commandes associé à l'uproc
CODE	Code auteur associé à l'utilisateur
CONDNO	Numéro de la condition
CPU	Consommation de temps CPU d'une exécution en secondes
CUM	Valeur cumulée
CYCLE	Cycle de l'échangeur en secondes
DALW	Plage horaire d'une tâche planifiée ou optionnelle pour tous les jours de la semaine
DAY	Jour dans la semaine
DEF	Fichier des valeurs par défaut d'un fichier d'ordres
DEP	Uproc conditionnante dans une condition d'enchaînement
DEPMU	Unité de gestion dépendante dans une relation TIH
DEV	Objet autorisé en univers de développement, contraire de NODEV
DIR	Nom du répertoire société
DISK	Nom du disque (sous VMS) du répertoire société
DDIR	Nom du répertoire des exécutables ou des applications en univers de développement
DDSK	Nom du disque (sous VMS) des exécutables ou des applications en univers de développement
DLNM	Nom logique (sous VMS) des exécutables ou des applications en univers de développement

DTRAIT	Date de traitement insérée dans le nom d'une ressource
ECH	Automate échangeur
EDATE	Date explicite de planification d'une tâche
EDT	Editeur utilisé sous VMS, autre valeur TPU
ELAPS	Consommation de temps elapsed d'une exécution en secondes
ENABLE	lors de l'insertion (uxins) d'une uproc ou d'une session, l'objet n'a pas besoin d'être libéré par l'option "06". L'action de libération sera nécessaire avec le qualifier NOENABLE
EXCLUD	Inversion logique du sens d'une condition (contraire de EXPECT)
EXLW	Dates et heures d'exclusion d'un lancement
EXECUTE	Vérification et exécution du fichier d'ordres d'une commande uxload, autre valeur VERIFY
EXPECT	Condition attendue (contraire de EXCLUD)
FATAL	Condition fatale, contraire de NOFATAL
FATHER	Uproc mère d'une branche de session
FNAME	Nom de fichier pour une ressource
FORCE	Indicateur de lancement forcé en fin de période d'une tâche ou d'un lancement, contraire de NOFORCE
FPERIOD	Période fonctionnelle d'une uproc
FREQ	Période d'examen (en secondes) d'une ressource (non logique) par l'automate surveillant
FRHOLI	Calendrier généré avec les jours fériés français, contraire de NOFRHOLI
FRI	Plage horaire de pilotage d'une tâche pour un vendredi ou valeur du jour dans la semaine
FULL	Permet l'affichage de toutes les informations liées à l'objet, autre valeur INF
HEADER	Uproc entête d'une session
HOLD	Tâche désactivée
HST	Affichage de la trace automate d'une exécution du suivi d'exploitation (Informations extraites de l'historique)
IDIR	Nom du répertoire data en espace intégration
IDSK	Nom du disque (sous VMS) en espace intégration
ILNM	Nom logique (sous VMS) en espace intégration
INC	Uproc conditionnante dans une condition de non simultanéité
INF	Informations techniques liées à l'objet considéré, autre valeur FULL
IODIR	Consommation statistique d'entrées / sorties directes de l'exécution
IOBUF	Consommation statistique d'entrées / sorties bufferisées de l'exécution
LABEL	Libellé descriptif de l'objet
LAN	Automate lanceur
LEND	Date et heure de fin de plage horaire d'un lancement
LFORM	Formule de lancement – formule booléenne combinant toutes les conditions d'une uproc
LNM	Nom logique (sous VMS) du répertoire société
LOCK	Verrouillage des uprocs et des **session**s lors des opérations de transfert ou de distribution ou bien Exclusivité sur les quotas d'une ressource, autre valeur UNLOCK
LOG	Affichage du fichier log d'une exécution du suivi d'exploitation ou bien ressource de type logique
LSTART	Date et heure de début de plage horaire d'un lancement
LW	Plage horaire de lancement pour la reprise d'une exécution
MASTER	Noeud maître
MAX	Valeur maximale sélectionnée dans les statistiques
MULW	Plage horaire d'une tâche multi journalière
MEMO	Mémorisation d'un événement d'exploitation pour une uproc

MIN	Valeur minimale sélectionnée dans les statistiques
MODEL	Objet modèle (calendrier ou tâche), autre valeur NOMODEL
MON	Plage horaire d'une tâche pour un lundi ou valeur du jour dans la semaine
MONI	Noeud de contrôle centralisé, autre valeur NOMONI
MULT	Horaires de lancement multiples d'une tâche, autre valeur NOMULT
NATURE	Nature d'une ressource : FIL, LOG, DSK, PRC, QUE ou SYS
NBEXEC	Numéro d'exécution statistique
NBPER	Nombre de périodes (fonctionnelle) sur lequel l'événement associé à une uproc est mémorisé
NO BYPASS	Lancement avec pilotage, contraire de BYPASS
NO CENTRAL	L'exécution ne sera pas remontée au noeud de contrôle centralisé, contraire de CENTRAL
NOCLEAR	Libération non automatique de la ressource en fin d'exécution de l'uproc, contraire de CLEAR
NODEV	Objet interdit en univers de développement, contraire de DEV
NOFATAL	Condition non fatale, contraire de FATAL
NOFORCE	Lancement non forcé en fin de plage horaire, contraire de FORCE
NO FRHOLI	Calendrier généré sans les jours fériés français, contraire de FRHOLI
NOHOLD	Tâche active, contraire de HOLD
NOMODEL	Objet non modèle, contraire de MODEL
NOMONI	Pas le noeud de contrôle centralisé, contraire de MONI
NOMULT	Horaires de lancement mono journaliers, contraire de MULT
NOPROD	Objet interdit en univers de production, contraire de PROD
NO RESTART	Indicateur de non relance automatique d'une tâche ou d'un lancement, contraire de RESTART
NOVERIF	Pas de vérification du fichier, contraire de VERIF
NOTIFY	Option de sélection sur la date
NSEQ	Numéro de séquence du lancement
NSESS	Numéro d'exécution de la session
NUPR	Numéro d'exécution de l'uproc
NSTAT	Nouvel état associé à l'événement
OFFSET	Report après calcul d'une règle de planification
ONE	Au moins une unité de gestion visée est nécessaire, autre valeur ALL
OPER	Opérateur dans la vérification de l'attribut d'une condition de ressource
PAGEF, PGF	Consommation statistique de page fault de l'exécution
PDATE	Contrôle ou calcul de la date de traitement
PDIR	Nom du répertoire des exécutables ou applications en univers de production
PDSK	Nom du disque (sous VMS) des exécutables ou applications en univers de production
PLNM	Nom logique (sous VMS) des exécutables ou applications en univers de production
PERIOD	Période d'une règle de planification
POS	Positionnement en nombre de jours dans la période de la règle
PRINTER	Imprimante associée à l'exécution d'une tâche ou d'un lancement
PRLW	Plage horaire de lancement d'une tâche provoquée
PROD	Objet autorisé en univers de production, contraire de NOPROD, autres valeur DEV et ALL
PROF	Profil de l'utilisateur
PRTY	Priorité d'exécution d'une tâche ou d'un lancement
QT1, QT2	Quotas d'une ressource (maximum ou alloués)

QUEUE	Queue batch d'exécution d'une tâche ou d'un lancement
REPL	Option de remplacement
RESTART	Indicateur de relance automatique d'une tâche ou d'un lancement, contraire de NORESTART
RULE	Règle et date d'application de la règle affectée à une tâche
SAT	Plage horaire d'une tâche pour un samedi ou valeur du jour dans la semaine
SDATE	Date de première planification d'une tâche
SESS	Contrôle inter session d'une condition
SDIR	Nom du répertoire data en espace simulation
SDSK	Nom du disque (sous VMS) en espace simulation
SLNM	Nom logique (sous VMS) en espace simulation
SINCE	Date de début de la période considérée ou première année du calendrier
SONOK	Uproc fille en chemin normal d'une session
SONKO	Uproc fille en chemin anormal d'une session
STATUS	Etat recherché d'une condition ou état du lancement, de l'exécution ou de l'événement
STEP	Numéro de jalon de l'événement ou de la reprise
SUN	Plage horaire d'une tâche pour un dimanche ou valeur du jour dans la semaine
SURV	Automate surveillant
TERM	Consigne de terminaison
TERMNO	Numéro de la consigne de terminaison
TESP	Code de l'espace cible du transfert (A, I, S, X)
THU	Plage horaire d'une tâche pour un jeudi ou valeur du jour dans la semaine
TLABEL	Libellé de l'objet résultat de la duplication
TMU	Unité de gestion cible de la distribution
TMUT	Type d'unité de gestion cible de la distribution
TMODEL	Tâche dupliquée modèle
TNOMODEL	Tâche dupliquée non modèle
TO	Nouveau type d'un jour
TPU	Editeur utilisé sous VMS, autre valeur EDT
TUE	Plage horaire d'une tâche pour un mardi ou valeur du jour dans la semaine
TUPR	Code uproc résultat de la duplication
TVUPR	Version d'uproc résultat de la duplication
TSES	Code session résultat de la duplication
TVSES	Version de session résultat de la duplication
TYPE	Type d'une tâche (parmi S, O et P)
UNLOCK	Non verrouillage des uprocs et des sessions lors des opérations de transfert ou de distribution ou bien non exclusivité sur les quotas d'une ressource, autre valeur LOCK
VALUE	Valeur de comparaison de l'attribut d'une condition de ressource
VERIF	Vérification du fichier de la ressource lors du pilotage de l'uproc, contraire de NOVERIF
VERIFY	Vérification et non exécution du fichier d'ordres, autre valeur EXECUTE
WED	Plage horaire d'une tâche pour un mercredi ou valeur du jour dans la semaine
XCLASS	Classe d'incompatibilité de l'uproc
XDATE	Date d'interdiction de la planification d'une tâche
XDIR	Nom du répertoire data en espace exploitation
XDSK	Nom du disque (sous VMS) en espace exploitation
XLNM	Nom logique (sous VMS) en espace exploitation

Exemple d'utilisation des commandes du CL

Les jalons de reprise et les messages dans un script UNIX ksh

```
#!/bin/ksh
STEP0 ()
# Variable S_NUMJALON  = 0 at normal start of uproc
{
$UXEXE/uxset msg "   *** Passage STEP 0 *** "
}
STEP1 ()
{
$UXEXE/uxset step "1"
$UXEXE/uxset msg "   *** Passage STEP 1 *** "
# Use uxset msg to display normal comments and
# instructions in History Trace.
}
STEP2  ()
{
$UXEXE/uxset step "2"
$UXEXE/uxset msg "   *** Passage STEP 2 *** "
# Utililiser uxset msg pour afficher dans la trace
# historique les commentaires et instructions
# à l'operateur.
}
STEP3 ()
{
$UXEXE/uxset step "3"
$UXEXE/uxset msg "   *** Passage STEP 3 *** "
}
STEP4 ()
{
$UXEXE/uxset step "4"
$UXEXE/uxset msg "   *** Passage STEP 4 *** "
}
STEP5 ()
{
$UXEXE/uxset step "5"
$UXEXE/uxset msg "   *** Passage STEP 5 *** "
}
# Lors d'une reprise la variable S_NUMJALON est alimentée manuellement à
partir de l'écran de suivi
# d'exploitation. Sinon = 0
# During recovery the S_NUMJALON variable is manually loaded from the Job
Monitor screen, else=0
#
Iscn=$S_NUMJALON

let Iscn=Iscn
echo "Iscn = $Iscn"
while [ $Iscn -le 5 ]
do
   echo "Iscn = $Iscn"
   Fnct=STEP$Iscn
   echo "Fnct = $Fnct"
   eval $Fnct
```

```
   let Iscn=Iscn+1
done
exit 0
```

Les steps de reprise et les messages dans un script Unix CSH.

```
#!/bin/csh
#Tronc commun de la procédure
goto STEP$S_NUMJALON

# point de démarrage normal
STEP00:

STEP01:
$UXEXE/uxset step "1"
$UXEXE/uxset msg "   *** Passage step N° $S_NUMJALON ***"

STEP02:
$UXEXE/uxset step "2"
$UXEXE/uxset msg "   *** Passage step N° $S_NUMJALON ***"

STEP99:
exit 0
```

Release notes

Corrections

Tableau de description des commandes : paragraphes "Les commandes de paramétrage" page 1 et "Les commandes d'exploitation" page 2.

Liste des commandes de gestion de batch : paragraphe "Les commandes de gestion de batch" page 2.

Liste des commandes de gestion de dates : paragraphe "Les commandes de gestion de date" page 2.

Corrections des commandes de paramétrage

L'item DAY= de la commande uxupd CAL prend la valeur ALL par défaut.

Corrections des commandes d'exploitation

L'item STATUS de la commande uxshw EVT permet une sélection des événements sur leur état.

Le suivi d'exploitation est désormais nommé "Exécutions".

Dans les commandes portant sur l'objet CTL, l'item OUTPUT est incompatible avec l'item LOG.

Corrections des commandes du CL d'exécution

Les variables du CL sont décrites en un endroit unique de la documentation utilisateur de Dollar Universe : le manuel de référence.

Dans la commande uxset var, si plusieurs variables doivent être transmises, elles doivent être séparées par des virgules : paragraphe "Reconduction de variables" page 131.

Modification des commandes

Modification des commandes de paramétrage v5.0 SPRE0025

Nouvel item NOVERIF dans la commande uxext (tous objets).

Famille GENINF obligatoire pour la commande uxadd UPR.

Les tâches non modèles peuvent faire l'objet d'extraction et d'insertion.

- L'item CIBLE de la commande uxins TSK peut prendre de nouvelles valeurs.

- Les tâches non modèles sont insérées par défaut.
- La tâche est insérée dans l'état Activée / Désactivée qu'elle avait lors de l'extraction.

Modification des commandes de paramétrage v5.1

Les anciens champs "Disque" et "Répertoire logique" ne doivent plus être utilisés par les commandes UXADD dans les champs Répertoires Société, UG, Application, Nœud, sous Windows, Unix et VMS.

Modification des commandes d'exploitation

La commande uxadd FLA affiche dans la sortie standard le numéro de lancement.

La commande uxadd FLA prend par défaut une fenêtre de lancement dont le début est "l'heure système courante" et la fin de fenêtre est "l'heure système courante plus 5 minutes".

La commande uxlst CTL accepte trois items supplémentaires : NOTIFY, MGT et VAR.

Les commandes UXLST ATM, UXSTR ATM et UXEND ATM sont "CLUSTER-WIDE" sous VMS : paragraphe "Commandes "Cluster-wide" sous VMS" page 106.

Modification des commandes de gestion de batch

Le comportement de la commande uxordre peut être paramétré par U_WRITE_HELD : paragraphe "Déclenchement de batch" page 115.

La commande uxordre affiche le numéro de lancement dans la sortie standard.

La commande uxordre accepte l'item BYPASS.

La commande uxsurjob prend en argument un temps en minutes (au lieu des secondes).

Modification des commandes de dates

Le résultat de la commande uxdat peut être en minuscules ou en majuscules selon la casse du format indiqué.

Le format de l'année de la commande uxdat peut être codé sur 2 caractères.

Nouvelles commandes

Nouvelles commandes de paramétrage

Commande de distribution :

- de la table société,
- de la table des nœuds,
- de la table des unités de gestion,
- de la table des dépendances des unités de gestion,
- de la table des domaines,
- de la table des applications,
- de la table des répertoires des applications,
- de la table des répertoires des unités de gestion,

- de la table des utilisateurs,
- des classes d'uprocs,
- des ressources,
- des uprocs,
- des sessions,
- des calendriers,
- des tâches.

Commande de transfert :

- des uprocs,
- des sessions,
- des calendriers,
- des tâches.

Commande de duplication :

- d'une uproc,
- d'une session,
- d'une tâche.

Commandes sur les profils utilisateurs : paragraphe "Les profils utilisateurs" page 32.

Commande pour définir la version courante d'une uproc.

Commande d'extraction et d'insertion de règles.

Commande pour libérer une uproc.

Nouvelles commandes d'exploitation

Commande pour supprimer des événements ressource.

Commandes sur l'historique des exécutions.

Commandes sur les statistiques.

Nouvelles commandes de gestion de batch

Exécuter un script utilisateur au-delà d'une certaine durée d'exécution de l'uproc : paragraphe "Surveillance de la durée d'exécution" page 118.

Nouvelles commandes de dates

Paragraphe "Formatage de date et décalage d'heure" page 124.

Nouvelles commandes du CL en v5.1

Paragraphe "Affichage d'informations" page 129.

Commandes VMS

Les anciennes commandes VMS avaient été remplacées en v5.0 par des commandes unifiées (Windows, UNIX et VMS). Elles avaient été maintenues pour des raisons de compatibilité.

Avec la version 5.1, la compatibilité a été pour certaines d'entre elles, voir le tébleau ci-dessous.

Commandes VMS abandonnées

U$ADD	Remplacée par **uxadd**
U$ASSIGN	Supprimée, obsolescence déclarée en version 3.4
U$COPY	Supprimée, obsolète depuis la version 3.4
U$CREPARAM	Remplacée par **uxset parm**
U$CREPARM	Supprimée, obsolète depuis la version 3.4
U$DELETE	Supprimée, obsolète depuis la version 3.4
U$DLT	Remplacée pat **uxdlt**
U$END	Remplacée par **uxend**
U$EXEC	Supprimée, obsolète depuis la version 3.4
U$EXT	Remplacée par **uxext**
U$GETPARM	Supprimée, obsolète depuis la version 3.4
U$HLD	Remplacée par **uxhld**
U$INS	Remplacée par **uxins**
U$INTER	Supprimée, obsolète depuis la version 3.4
U$LIST	Supprimée, obsolète depuis la version 3.4
U$MBX	Supprimée, obsolète depuis la version 3.4
U$MSG	Supprimée, obsolète depuis la version 3.4
U$PARAM	Supprimée, obsolète depuis la version 4.3
U$PARM	Supprimée, obsolète depuis la version 4.3
U$PROVOC	Supprimée, obsolète depuis la version 4.3
U$PSTAT	Supprimée, obsolète depuis la version 4.3
U$PURGE	Supprimée, obsolète depuis la version 4.3
U$RENAME	Supprimée, obsolète depuis la version 4.3
U$RENAME_40	Supprimée, obsolète depuis la version 4.3
U$RLS	Remplacée par **uxrls**
U$RST	Remplacée par **uxrst**
U$RTVDIR	Compatibilité provisoire par U$RTVREPER
U$SET	Remplacée par **uxset**
U$SHW	Remplacée par **uxshw**
U$SIM	Supprimée, obsolète depuis la version 4.3
U$STAT	Supprimée, obsolète depuis la version 4.3
U$STEP	Supprimée, obsolète depuis la version 4.3
U$STR	Remplacée par **uxstr**
U$SYMB	Supprimée, symboles générés dans coquille batch U_BATCH.COM
U$UNSET	Supprimée, obsolète depuis la version 4.3
U$UPD	Remplacée par **uxupd**
U$VRF	Supprimée, obsolète depuis la version 4.3

U$WAK	Remplacée par **uxset**

Commandes VMS maintenues provisoirement

U$MSGHPR	Compatibilité provisoire, remplacée par **uxset msg**
U$ORD	Compatibilité provisoire, remplacée par **uxord**
U$RTVREPER	Compatibilité provisoire
U$WRTJAL	Compatibilité provisoire, remplacée par **uxset step**
U$DATE	Compatibilité provisoire, remplacée par **uxdat**

Index

$

$U
  commandes 1
  commandes de gestion de batch 115
  commandes de gestion de date 121
  commandes du CL 127
  liste des commandes de gestion de batch 2
  liste des commandes de gestion de date 2
  liste des commandes de paramétrage 1
  liste des commandes d'exploitation 2
  liste des commandes du CL 3

A

ADM
  uxext uxins 10
Afficher
  un répertoire application 27
  un répertoire UG 30
APD
  uxadd uxupd uxdlt uxshw uxlst uxdst 27
APPL
  uxadd uxupd uxdlt uxshw uxlst uxdst 25
ATM
  uxend uxstr uxwak uxlst 105

B

Batch
  déclencher 115
  surveiller la durée d'exécution 118

C

CAL
  uxadd uxupd uxdlt uxshw uxlst uxtra uxdst uxext
    uxins 61
CL
  code retour 127
  commandes 128
  liste des commandes 3
  normes d'écriture des commandes 127
  présentation des commandes 127
CLASS
  uxadd uxupd uxdlt uxshw uxlst uxdst 36
Code retour
  d'une commande 5
Commandes
  code retour 5
  déclaration 3
  du CL 3
  du CL – normes d'écriture 127
  du CL - présentation 127
  environnement 4
  exploitation 2
  fichier d'ordre 5
  fichier résultat ou d'entrée 4
  format des dates 7
  format général 3
  gestion de batch 2
  gestion de batch - présentation 115
  gestion de date 2
  gestion de date - présentation 121
  items 4
  paramétrage 1
  présentation 1
  règles générales d'utilisation 5
  thème 4
  utilisation 5
  utilisation des guillemets 5
  verbe 3
Consignes
  écrire dans la trace automate 129
Créer
  un répertoire application 27
  un répertoire UG 29
Critères de gestion
  définition et valorisation 130
  format de la commande 131
  format du fichier de référence 130
CTL
  uxpur uxend uxrst uxlst 100

D

Date
  format dans une commande 7
  formatage ou décalage 121
Décalage
  d'une date 121
  d'une heure 124
Déclaration
  d'une commande 3
Déclencher une Uproc
  commande 115
Définition

critère de gestion 130
Distribuer
la table des répertoires application 28
la table des répertoires UG 30
DOM
uxadd uxupd uxdlt uxshw uxlst uxdst 23
Durée d'exécution
commandes pour surveiller 118

E

Ecrire
des informations dans la liste des exécutions 129
un jalon (ou step) 128
un message dans la trace automate 129
Environnement
d'une commande 4
Espace
d'une commande 4
EVT
uxadd uxupd uxdlt uxshw uxlst 96
Exécutions
afficher des informations utilisateur 129
Exemple
arrêter une exécution après un temps donné 118
commandes sur la société 15
commandes sur le planning prévisionnel 110
commandes sur le suivi d'exploitation 104
commandes sur le type d'UG 19
commandes sur les automates 106
commandes sur les dépendances d'UG 23
commandes sur les statistiques 113
commandes sur l'historique des exécutions 109
commandes sur un calendrier 65
commandes sur un domaine 25
commandes sur un événement 99
commandes sur un lancement 95
commandes sur un noeud 18
commandes sur un profil 33
commandes sur un répertoire application 29
commandes sur un répertoire UG 31
commandes sur un utilisateur 36
commandes sur une application 27
commandes sur une classe 37
commandes sur une règle 68
commandes sur une ressource 41, 87
commandes sur une session 60
commandes sur une tâche 80, 89
commandes sur une UG 21
commandes sur une Uproc 55
décalage de dates 124
décalage horaire 125
déclenchement d'une tâche provoquée 117
écrire des informations dans la liste des exécutions 129
écrire un jalon 128
écrire un message dans la trace automate 129
format de dates 123
format horaire 125
lancer un script après un temps donné 119, 120
passer des paramètres à une tâche provoquée 117
passer des paramètres dans la session 130
reconduire des variables dans une session 132
surveiller la durée d'exécution d'une uproc 118, 119, 120
utiliser les critères de gestion 131
uxext uxins ADM 12

F

Familles
STA 112
TSK 75
UPR 49
Fichier
résultat ou entrée d'une commande 4
Fichier d'ordre
utilisation des commandes 5
FLA
uxadd uxupd uxdlt uxshw uxlst uxhld uxrls 90
Formatage
modifier une date 121
modifier une date et heure 124

G

Gestion de batch
présentation des commandes 115
Gestion de date
présentation 121
Guillemets
dans une commande 5

H

HCX
uxshw uxlst 106
HDP
uxadd uxdlt uxshw uxlst uxdst 22
Heure
formatage ou décalage 124

I

Items
ADM 11
APD 28
APPL 26
ATM 105
CAL 63
CLASS 37
CTL 102, 107

DOM 24
d'une commande 4
EVT 98, 112
FLA 93
HDP 23
MU 21
MUD 31
MUT 19
NODE 17
PROFIL 33
PRV 109
RES 40, 85
RUL 68
SES 59
SOC 14
TSK 76, 89
UPR 49
USER 35
uxalrjob 120
uxdat 122
uxload 9
uxordre 116
uxset info 129
uxset mgt 131
uxset msg 129
uxset parm 130
uxset step 128
uxset var 132
uxspvjob 119
uxsurjob 118
uxtim 124

J

Jalon
écriture 128

L

Lister
les répertoires application 28
les répertoires UG 30

M

Message
écrire dans la trace automate 129
Modifier
un critère de gestion 130
un répertoire application 27
un répertoire UG 29
MU
uxadd uxupd uxdlt uxshw uxlst uxdst 19
MUD
uxadd uxupd uxdlt uxshw uxlst uxdst 29
MUT
uxadd uxupd uxdlt uxshw uxlst 18

N

NODE
uxadd uxupd uxdlt uxshw uxlst uxdst 15
Noeud
d'une commande 4

P

Paramètres
d'une commande 4
passer à une tâche provoquée 115
passer dans la session 130
PROF
uxadd uxupd uxdlt uxlst uxext uxins 32
Provoquer une tâche
commande 115
PRV
uxlst 109

R

Reconduire
des variables dans une session 131
Règles générales
d'utilisation d'une commande 5
Répertoire application
commandes de manipulation 27
Répertoire UG
commandes de manipulation 29
RES
uxadd uxupd uxdlt uxshw uxlst uxext uxins uxdst 38
uxhld uxrls uxshw uxrsv uxcnr uxpre 83
RUL
uxadd uxupd uxdlt uxshw uxlst uxext uxins 66

S

SES
uxadd uxdup uxdlt uxshw uxlst uxtra uxdst uxext uxins 56
Session
passer des paramètres 130
reconduction des variables 131
SOC
uxadd uxupd uxdlt uxshw uxlst uxdst 13
Société
d'une commande 4
STA
uxadd uxupd uxdlt uxshw uxlst 110
Step
écriture 128
Supprimer
un répertoire application 27

un répertoire UG 30
Surveiller la durée d'exécution
commandes 118

T

Tâche
déclencher 115
surveiller la durée d'exécution 118
Thème
d'une commande 4
Trace automate
écrire un message 129
TSK
uxadd uxupd uxdup uxdlt uxshw uxlst uxtra uxdst uxext uxins 69
uxhld uxrls uxlst 88

U

UPR
uxadd uxupd uxdup uxdlt uxshw uxlst uxrls uxscv uxtra uxdst uxext uxins 42
Uproc
déclencher 115
passer des paramètres dans la session 130
reconduire des variables dans la session 131
surveiller la durée d'exécution 118
USER
uxadd uxupd uxdlt uxshw uxlst uxdst 34
Utilisation
d'une commande 5
uxadd
APD 27
APPL 25
CAL 61
CLASS 36
DOM 23
EVT 96
FLA 90
HDP 22
MU 19
MUD 29
MUT 18
NODE 15
PROF 32
RES 38
RUL 66
SES 56
SOC 13
STA 110
TSK 69
UPR 42
USER 34
uxalrjob
fonction 119
format 119
uxcnr
RES 84
uxdat
fonction 121
format 121
uxdlt
APD 27
APPL 26
CAL 62
CLASS 36
DOM 24
EVT 97
FLA 91
HDP 22
MU 20
MUD 30
MUT 18
NODE 16
PROF 32
RES 39
RUL 67
SES 57
SOC 14
STA 111
TSK 73
UPR 47
USER 34
uxdst
APD 28
APPL 26
CAL 63
CLASS 37
DOM 24
HDP 22
MU 20
MUD 30
NODE 16
RES 39
SES 58
SOC 14
TSK 74
UPR 47
USER 35
uxdup
SES 56
TSK 72
UPR 46
uxend
ATM 105
CTL 100
uxext
ADM 10
AUT PROFIL 32
CAL 63
RES 39

RUL 67
SES 58
TSK 74
UPR 48
uxhld
FLA 91
RES 83
TSK 88
uxins
ADM 11
AUT PROFIL 33
CAL 63
RES 40
RUL 67
SES 58
TSK 75
UPR 48
uxload
fonction 9
format 9
items 9
uxlst
APD 28
APPL 26
ATM 105
CAL 62
CLASS 37
CTL 101
DOM 24
EVT 97
FLA 92
HCX 107
HDP 22
MU 20
MUD 30
MUT 19
NODE 16
PROF 32
PRV 109
RES 39
RUL 67
SES 57
SOC 14
STA 112
TSK 73, 88
UPR 47
USER 35
uxordre
fonction 115
format 116
uxpre
RES 85
uxpur
CTL 100
uxrls
FLA 91
RES 83
TSK 88
UPR 48
uxrst
CTL 101
uxrsv
RES 84
uxscv
UPR 48
uxset
info 129
mgt 130, 131
msg 129
parm 130
step 128
var 131
uxshw
APD 27
APPL 25
CAL 61
CLASS 36
DOM 24
EVT 96
FLA 92
HCX 106
HDP 22
MU 20
MUD 30
MUT 18
NODE 16
RES 38, 84
RUL 67
SES 57
SOC 13
STA 111
TSK 73
UPR 46
USER 34
uxspvjob
fonction 119
format 119
uxstr
ATM 105
uxsurjob
fonction 118
format 118
uxtim
fonction 124
format 124
uxtra
CAL 62
SES 57
TSK 74
UPR 47
uxupd
APD 27

APPL 25
CAL 61
CLASS 36
DOM 23
EVT 96
FLA 90
MU 20
MUD 29
MUT 18
NODE 15
PROF 32
RES 38
RUL 66
SOC 13
STA 110
TSK 70
UPR 42
USER 34
uxwak
ATM 105

V

Variables
reconduction dans la session 131
Verbe
d'une commande 3

Email : support@orsyp.com
Site WEB : http://www.orsyp.com